Sommario

341-342 - novembre-dicembre 2013

tutorial

progetti

rubriche

Editoriale

Giochi di luce...

Ricordo ancora molto bene i miei primi esperimenti con l'elettronica da bambino. Tra i vari esperimenti finiti male, ne ricordo uno in particolare con i LED, nato da una considerazione da completo neofita e inesperto: con una batteria da 1,5V accendo un LED, poi con due batterie in serie accendo lo stesso LED che, malgrado la sua "sofferenza" non si danneggia e si accende con una luminosità maggiore. Da qui l'insana idea di collegarlo alla rete a 240Vac per vedere quanta luminosità riusciva ad emettere quel componente. Inutile dire che il tutto è finito con "gioco di luci e fumo" con conseguente disintegrazione del povero LED. Un esperimento assolutamente da non fare ma la curiosità spesso porta a decisioni malsane. Per evitare la brutta esperienza ai nostri giovani lettori, abbiamo elaborato, in questo numero, un articolo speciale dedicato proprio ai LED. Una breve introduzione al funzionamento, i circuiti di polarizzazione e numerosi circuiti applicativi via via sempre più complessi. Potrete così sperimentare in tutta sicurezza con questo piccolo componente dalle molteplici applicazioni. I più esperti troveranno il progetto di un dimmer a microcontrollore, un inverter per ottenere una tensione di 240Vac a partire da una batteria, un interessante impiego degli accelerometri per costruire un dispositivo di puntamento e molti altri progetti interessanti. Un numero natalizio adatto a tutti, con il quale tutto il nostro Staff vi augura Buone Feste ed un brillante 2014!

Maurizio Del Corso

DIRETTORE RESPONSABILE
Antonio Cirella

DIRETTORE TECNICO
Maurizio Del Corso

Segreteria di redazione
Giorgia Generali

Comitato scientifico
Simone Masoni (Microtest), Francesco Picchi (Microtest), Massimo Rovini (Università degli Studi di Pisa).

Grafica
Studio Punto & Virgola

Ha collaborato a questo numero
Angelo De Bartolo

Direzione e redazione
INWARE srl
Via Giotto, 7 - 20032 Cormano (MI)
Tel. 02.66504794
Fax 02.42101817
info@inwaredizioni.it
www.inwaredizioni.it
Redazione: fe@inwaredizioni.it

Pubblicità per l'Italia
Agostino Simone
Tel. +39 347 2230684
media@inwaredizioni.it

Europe and Americas
Elisabetta Rossi
Tel. +39 328 3245956
international@inwaredizioni.it

Asia
Cybermedia Communications Inc.
Tel. +886-(0)2-2691-2785
asia@inwaredizioni.it

Rest of the world
Inware srl
+39 02 66504794
info@inwaredizioni.it

Ufficio abbonamenti
INWARE srl
Via Giotto, 7 - 20032 Cormano (MI)
Per informazioni, sottoscrizione o rinnovo dell'abbonamento:
abbonamenti@inwaredizioni.it
Tel. 02.66504794
Fax. 02.42101817
L'ufficio abbonamenti è disponibile telefonicamente dal lunedì al venerdì dalle 14,30 alle 17,30.

• Membership MAKER:
11 numeri di Fare Elettronica in pdf € 29.99
• Membership MAKER FULL:
11 numeri di Fare Elettronica in pdf + Bonus Pack Maker € 39.99

Autorizzazione alla pubblicazione
Tribunale di Milano n.647
del 17/11/2003

Privacy
Nel caso la rivista sia pervenuta in abbonamento o in omaggio, si rende noto che i dati in nostro possesso sono impiegati nel pieno rispetto del D.Lgs. 196/2003. I dati trasmessi a mezzo cartoline o questionari presenti nella rivista, potranno venire utilizzati per indagini di mercato, proposte commerciali, o l'inoltro di altri prodotti editoriali a scopo di saggio. L'interessato potrà avvalersi dei diritti previsti dalla succitata legge. In conformità a quanto disposto dal Codice di deontologia relativo al Trattamento di dati personali art. 2, comma 2, si comunica che presso la nostra sede di Cormano Via Giotto 7, esiste una banca dati di uso redazionale. Gli interessati potranno esercitare i diritti previsti dal D.Lgs. 196/2003 contattando il Responsabile del Trattamento Inware Edizioni Srl (info@inwaredizioni.it).

Collaborare con FARE ELETTRONICA
Le richieste di collaborazione vanno indirizzate all'attenzione di Maurizio Del Corso (m.delcorso@inwaredizioni.it) e accompagnate, se possibile, da una breve descrizione delle vostre competenze tecniche e/o editoriali, oltre che da un elenco degli argomenti e/o progetti che desiderate proporre.

Elenco inserzionisti

Teledyne LeCroy
via E. Mattei Valecenter 1/C - 30020 Marcon (VE)
Tel. 041/5997011 - www.lecroy.com

PCB Pool
Bay 98-99 - Shannon Free Zone -
Shannon - County Clare
Tel. +353 (0)61 701170 – www.beta-layout.com

Micromed
Via Valpadana 126B/2 - 00141 Roma (RM)
Tel. 06/90024006 - www.micromed.it

Grifo
Via dell'Artigiano 8/6 -
40016 San Giorgio Di Piano (BO)
Tel. 051-892052 - www.grifo.it

Mesago Messe Frankfurt
Rotebuehlstr. 83 - 85 - 70178 Stuttgart,
Tel. +49 711 61946-89 - www.mesago.com/sps

Millennium Dataware
Corso Repubblica 48 - 15057 Tortona (AL)
Tel. 0131 860254 - www.mdsrl.it

MikroElektronika
Visegradska, 1A - 11000 Belgrade
Tel. +381 11 3628830 - www.mikroe.com

AFRA - Associazione Fra Radioamatori Abruzzo
Via delle Fornaci 2 - 65125 Pescara (PE)
Tel. 085/4714835 - www.aripescara.org

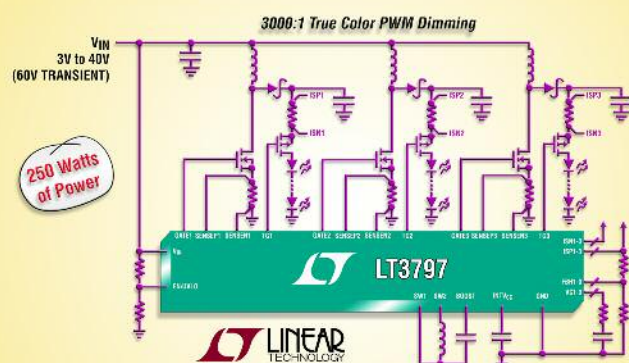
Elettroshop
Via Giotto, 7 - 20032 Cormano (MI)
Tel. 02 66504755 - www.elettroshop.com

Tecnoimprese
Via Console Flaminio, 19 - 20134 (MI)
Tel. 02 2101111 – www.fortronic.it

News

Controller per driver LED ad alta corrente e a tre uscite

Linear Technology Corporation annuncia l'LT3797, un controller DC/DC a tre uscite per gestire tre canali di LED indipendenti. L'architettura in current mode e a frequenza fissa del dispositivo fornisce corrente LED costante e con regolazione precisa in un ampio range di tensioni di alimentazione e uscita. L'intervallo di tensioni di ingresso da 2,5V a 40V con tensione transitoria fino a 60V lo rende ideale per le applicazioni automotive che devono funzionare in condizioni di avviamento afreddo, stop/start e load dump, oltre che per un'ampia gamma di applicazioni industriali. Ciascuno dei canali dell'LT3797 può gestire un'ampia gamma di potenza LED nelle topologie step-up, step-down o SEPIC. In una configurazione boost, alimenta tre canali fino a 90V di LED con correnti LED superiori a 1A, fornendo oltre 250W di potenza da un ingresso nominale di 12V con efficienza fino al 93%. Ciascuno dei tre canali è azionato da un segnale True Color PWM indipendente che ne consente il dimming con rapporti fino a 3000:1. Ciascun canale offre inoltre protezione contro l'apertura dei LED e i cortocircuiti in modalità boost, fornendo progetti ad alta sicurezza e affidabilità richiesti nelle applicazioni automotive. Grazie a un pin di regolazione della frequenza, l'utente può programmare la frequenza tra 100kHz e 1MHz in modo da ottimizzare il rendimento e ridurre al tempo stesso la dimensione dei componenti esterni. Il package QFN da 7 x 8mm con caratteristiche termiche avanzate fornisce una soluzione compatta e a ingombro ridotto per applicazioni LED da 20W a 250W.

L'LT3797 è progettato in modo tale che ciascun canale possa utilizzare la configurazione più adatta per gestire la rispettiva striscia di LED, che si tratti di step-up, step-down e SEPIC oppure una loro combinazione. La funzione di rilevamento della corrente sul lato alto consente di utilizzare ciascun canale in una configurazione uguale o diversa, mentre il rilevamento della corrente rail-to-rail consente uscite da 0V a 100V, offrendo ampia flessibilità di progettazione. Altre funzionalità includono blocco per sotto- o sovratensione, rapporto di dimming analogico di 20:1 tramite il pin CTRL e sincronizzazione esterna.

www.linear.com/product/LT3797

News

Microcontrollore per integrare il controllo dei motori e del sistema

Toshiba Electronics Europe ha annunciato la disponibilità di un nuovo microcontrollore basato su nucleo di elaborazione ARM CortexTM M3, che è in grado di controllare sia i motori elettrici, sia il sistema nel suo complesso. Il chip TMPM384FDFG integra in un singolo dispositivo una memoria di grande capacità, affiancata da temporizzatori multifunzione, convertitori AD e porte di I/O potenziate.
La applicazioni di controllo dei motori elettrici vengono di solito realizzato abbinando un'unità dedicata al controllo motore e un'unità di controllo del sistema, entrambe le quali utilizzano un microcontrollore e il relativo software. Il circuito integrato TMPM384FDFG permette di sfruttare un solo microcontrollore e relativo software per occuparsi di entrambe le funzioni, facilitando la attività di sviluppo del sistema rendendole più efficienti e riducendo i costi di produzione complessivi. Il chip TMPM384FDFG contiene 512 Kbyte di memoria flash e 32 Kbyte di memoria SRAM e può essere alimentato con una tensione di alimentazione singola a 5 V, un valore tipico dei dispositivi di controllo motore utilizzati in elettrodomestici e nelle apparecchiature industriali. Grazie alla disponibilità di convertitori AD a 22 canali da 12 bit estremamente accurati, porte di I/O con funzionalità avanzate e temporizzatori multifunzione a 4 canali, la nuova serie di microcontrollori TX03 permette di controllare sia i motori che gli IGBT.

www.toshiba-components.co

Un Companion Device BLDC trifase da Microchip

Microchip annuncia un nuovo motor gate driver BLDC trifase con modulo di potenza: l'MCP8024. Questo nuovo dispositivo include funzioni che forniscono al DSC dsPIC e microcontroller PIC la capacità di pilotare sei MOSFET N-channel. I clienti possono implementare prestazioni migliorate ed una elevata solidità che offre una più elevata efficienza, riducendo i costi di sistema e il time to market. L'MCP8024 funziona in un range di tensioni molto ampio tra 6V e 28V e può resistere a tensioni transienti fino a 48V. Il dispositivo fornisce elevata integrazione analogica, come, ad esempio, tre amplificatori operazionali per rilevamento di corrente; un compara-

News

tore di sovracorrente; driver per MOSFET e interfaccia di comunicazione bidirezionale per design di sistemi motori completi. Gestione del dead time del driver configurabile, controllo del blanking-time del driver e Over-Current Limit per MOSFET esterni offrono un significativo incremento della flessibilità. Il convertitore step-down DC-to-DC regolabile alimenta una vasta gamma di microcontroller con i vantaggi di efficienza di un alimentatore switch-mode. L'MCP8024 è supportato dal MCP8024 TQFP BLDC Motor Driver Evaluation Board di Microchip.

www.microchip.com/get/RTAS

Teledyne LeCroy aggiunge capacità di analisi di Segnali misti agli oscilloscopi ad alta deginizione della serie HDO

Teledyne LeCroy introduce sul mercato gli oscilloscopi ad alta definizione della serie HDO4000-MS e HDO6000-MS che combinano 16 canali aventi flessibili capacità sui segnali misti con la tecnologia ad alta definizione HD4096, una lunga memoria di acquisizione, dimensioni compatte e disponibolità di bande passanti da 200 MHz fino a 1 GHz. Tutti i modelli HDO sono dotati di ampio schermo Touch Screen da 12,1" e una interfaccia molto intuitiva che migliora l'esperienza operativa. Potenti strumenti di collaudo, una piena dotazione di misurazioni automatiche e capacità di calcolo sulle forme d'onda rendono gli osciloscopi HDO-MS una completa macchina per il collaudo di segnali analogici, digitali e seriali. Vista la crescente complessità dei sistemi embedded avere potenti capacità di collaudo su segnali misti diviene essenziale negli oscilloscopi moderni. Gli oscilloscopi HDO-MS vengono forniti con specifici strumenti di collaudo digitale uniti a 16 canali digitali integrati. Il Triggering Cross Pattern analogico e digitale, le misurazioni di temporizzazione digitale, la ricerca di Pattern paralleli, l'emulazione del Gate logico e gli indicatori di attività sono strumenti ideali per la precisa validazione ed il rapido collaudo di sistemi embedded complessi. Queste caratteristiche vanno ad aggiungersi all'ampia dotazione di strumenti tipici di HDO che includono lo strumento di ricerca WaveScan, il playback di forme d'onda in modalità History,l'innovativo Triger "TriggerScan" (HDO6000 serie), la modalità di acquisizione di sequenze e la generazione di rapporti LabNotebook. La disponibilità di pacchetti software per l'analisi di Spettro e le misurazioni Power espandono le capacità dell'oscilloscopio facendo leva sulla maggiore precisione di misurazione e il range dinami-

News

co più ampio tipici dell'architettura a 12 bit reali. Gli oscilloscopi HDO4000-MS sono caratterizzati da una frequenza di campionamento di 2,5 GS/s con 12.5 Mpts di memoria per ciascun canale e fino a 50 Mpts per canale in via Opziolale in modelli a 2 e 4 canali con bande passanti che vanno da 200 MHz fino a 1 GHz. Gli oscilloscopi HDO6000-MS a 4 canali sono caratterizzati da una frequenza di campionamento di 2,5 GS/s con 50 Mpts di memoria per ciascun canale nella configurazione standard e fino a 250 Mpts per canale in via opzionale. con bande passanti disponibili da 350 MHz, 500 MHz e 1 GHz.

http://teledynelecroy.com/europe/

Circuiti a bassissima potenza per progetti di energy harvesting di nuova generazione

Texas Instruments ha presentato cinque circuiti integrati di nuova generazione per la gestione di potenza, capaci di catturare e gestire in maniera efficiente quantità variabili di energia, da microwatt (uW) a milliwatt (mW), da sorgenti di luce, calore o energia meccanica. I modelli bq25570, bq25505, TPS62740, TPS62737 e TPS62736 assicurano i livelli di corrente quiescente più bassi del settore e consentono il funzionamento senza batteria di reti di sensori wireless, sistemi di monitoraggio, dispositivi medicali indossabili, accessori portabili e altre applicazioni con accesso limitato a fonti di energia.

www.ti.com

Cree presenta il nuovo modulo LMH2 con la prima tecnologia di sunset dimming del settore

Cree , Inc. introduce il modulo a LED LMH2 con sunset dimming, che offre la tecnologia di regolazione del flusso luminoso e cromaticità simile alle lampade ad incandescenza. Il nuovo apparecchio permette un profilo di regolazione naturale, che in precedenza non era raggiungibile con qualsiasi altra tecnologia a risparmio energetico. La tecnologia sunset dimming di Cree permette di variare gradatamente la luce da 2700K a 1800K per applicazioni che richiedono uno stile tradizionale di oscuramento, quali l'ospitalità e il settore residenziale, raggiungendo circa l'80% di risparmio energetico rispetto alle lampade ad incandescenza tradizionali. La vasta gamma di moduli LMH2 ora consente ai produttori di illuminazione di affrontare ulteriori applicazioni. Disponibili con diversi lumen output (da 850lm a 3000lm), i moduli LMH2 con tecnologia di sunset dimming possono essere accoppiati con i già esistenti driver utilizzati per le versioni standard e sono compatibili anche con i dimmer TRIAC (il modulo ad alto rendimento può essere facilmente installato in qualsiasi ambiente con tecnologie di oscuramento tradizionali o utilizzato in nuove installazioni con driver 0-10V) o con tecnologia di controllo DALI (Digital Addressable Lighting Interface).

www.cree.com/modules

Eventi

RADIANT

Dal 2002 la periodicità è divenuta quadrimestrale con appuntamenti fissi nei mesi di gennaio, giugno e ottobre. In un'animata carrellata tra passato e futuro, la Rassegna si configura in tre settori:
MOSTRA-MERCATO (apparati e componenti per telecomunicazioni, Internet e ricetrasmissioni di terra e satellitari, antenne, elettronica, informatica, telefonia statica e cellulare, hobbistica ed editoria)
BORSA-SCAMBIO (surplus radioamatoriale, informatico ed elettronico)
RADIOANTIQUARIATO (radio d'epoca, grammofoni, valvole, ricambi, dischi in vinile, schemari ed editoria specializzata).
Attualmente RADIANT, che si sviluppa su una superficie espositiva superiore ai 10.000 mq. coperti, con più di 180 espositori per edizione ed oltre 40.000 visitatori annui, è considerata la più importante Mostra-Mercato nazionale del suo settore. Fra le categorie merceologiche ricordiamo: elettronica, informatica, telefonia, radiantismo, editoria, tv satellitare, hobbistica, surplus, radio d'epoca

Dove: Novegro (MI) • Quando: 25-26 Gennaio 2014 • Orari: dalle 9.00 alle 18.00 • Organizzazione: Comis • info: www.parcoesposizioninovegro.it

MONDO ELETTRONICA RAVENNA

Nella suggestiva cornice del PALAZZO MAURO DE ANDRE', il polo fieristico della città di Ravenna, che ospita fiere, mostre-mercato e meeting nazionali, SABATO 1 e DOMENICA 2 FEBBRAIO si svolgerà il prestigioso appuntamento dedicato al mondo dell' elettronica, dell'informatica, della telefonia, del digitale, etc.

Dove: Ravenna • Quando: 1-2 Febbraio 2014 • Orari: dalle 9.00 alle 18.00 • Organizzazione: Exposition Service • info: www.mondoelettronica.net

EXPOELETTRONICA MODENA

Appuntamento con la 16esima edizione di Expo Elettronica a Modena Fiere, sabato 18 e domenica 19 gennaio 2014. Si tratta di una delle rassegne più complete ed interessanti per gli appassionati di elettronica professionale e di consumo a cui prendono parte più di trecento espositori con migliaia di prodotti proposti al pubblico a prezzi eccezionali.

Dove: Modena • Quando: 18-19 Gennaio 2014 • Orari: dalle 9.00 alle 18.00 • Organizzazione: Blunautilus • info: www.expoelettronica.it

vuoi far parte del CLUB? Sei MAKER, SMART o GENIUS?

Entra anche tu nel mondo dell'elettronica con Inware Edizioni!

CLUB MEMBERSHIP

Club Light
accesso e/o download agli articoli (oltre 300) del Club di Firmware

Club Full
accesso e/o download agli articoli (oltre 700) del Club di Fare Elettronica

SMART MEMBERSHIP

- 11 nuovi numeri della rivista digitale Firmware (i numeri andranno in coda all'eventuale abbonamento esistente)
- tutto l'archivio delle riviste Firmware pubblicate mensilmente dal febbraio 2010

Bonus Pack Smart (solo per la card Smart Full)

- accesso e/o download agli articoli (oltre 300) del Club di Firmware
- un ebook a scelta* (Smartcard, CPLD, Display LCD, PIC, Operazionali, alimentatori, linguaggio C, Basic per PIC, Elettronica Analogica)
- una raccolta delle annate di Firmware in PDF (dal 2006 al 2012)

GENIUS MEMBERSHIP

- 11 nuovi numeri di Fare Elettronica in PDF
- 11 nuovi numeri della rivista digitale Firmware (i numeri andranno in coda all'eventuale abbonamento esistente)
- tutto l'archivio delle riviste Firmware pubblicate mensilmente dal febbraio 2010

Bonus Pack Genius (solo per la card Genius Full)

- accesso e/o download agli articoli (oltre 300) del Club di Firmware
- accesso e/o download agli articoli (oltre 700) del Club di Fare Elettronica
- due ebook a scelta* (Smartcard, CPLD, Display LCD, PIC, Operazionali, alimentatori, linguaggio C, Basic per PIC, Elettronica Analogica)
- due raccolte delle annate di Fare Elettronica in PDF (dal 2003 al 2012) e Firmware (dal 2006 al 2012).

MAKER MEMBERSHIP

- 11 nuovi numeri di Fare Elettronica in PDF (i numeri andranno in coda all'eventuale abbonamento esistente)

Bonus Pack Maker (solo per la card Maker Full)

- accesso e/o download agli articoli (oltre 700) del Club di Fare Elettronica
- un ebook a scelta* (Smartcard, CPLD, Display LCD, PIC, Operazionali, alimentatori, linguaggio C, Basic per PIC, Elettronica Analogica)
- una raccolta delle annate di Fare Elettronica in PDF a scelta (dal 2003 al 2012)

SCEGLI LA TUA MEMBERSHIP E FARAI PARTE ANCHE TU DEL CLUB PIÙ ESCLUSIVO DELL'ELETTRONICA!

I prezzi delle card si intendono IVA inclusa

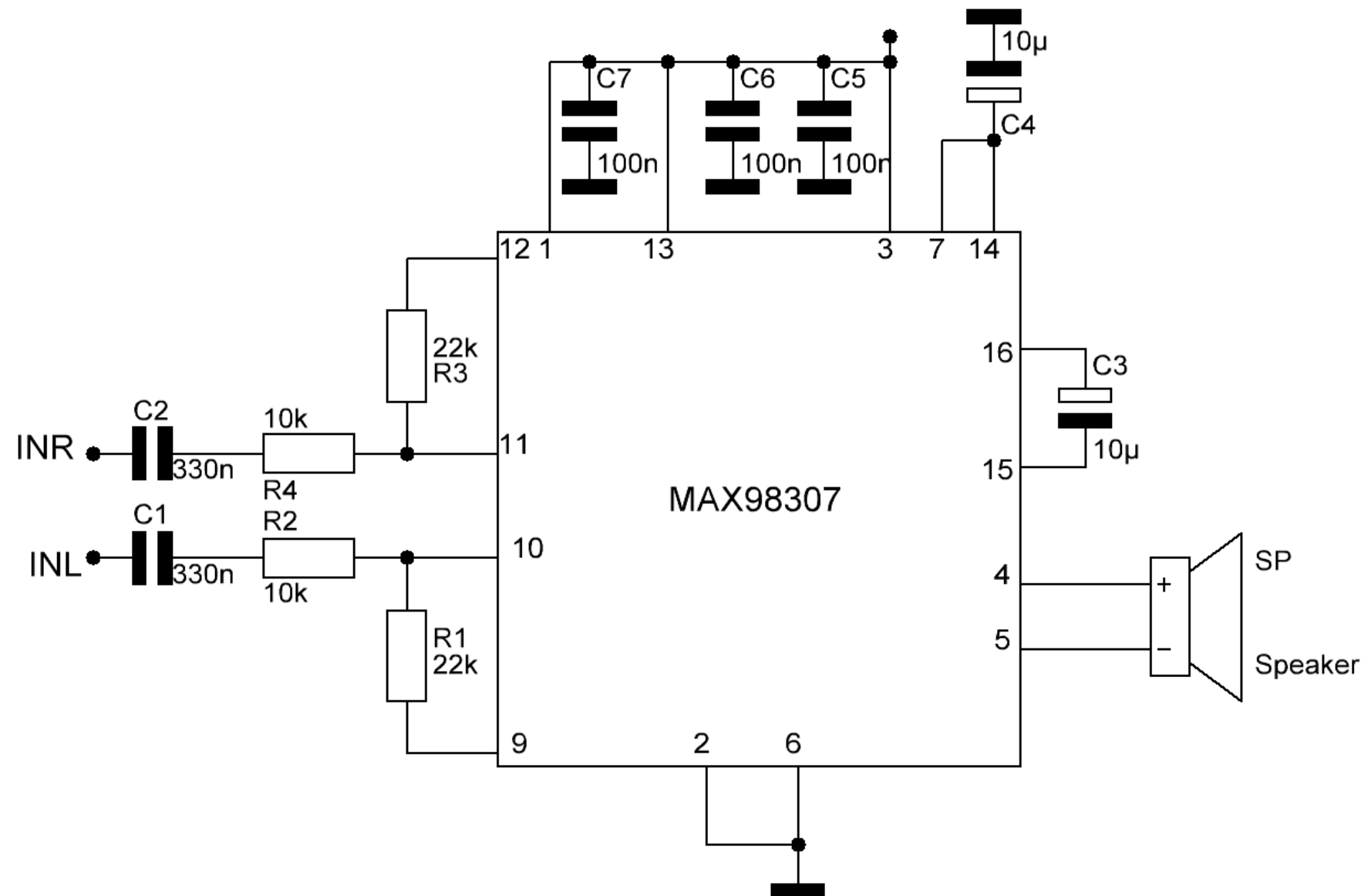
*la scelta viene effettuata al ricevimento della card

Idee di progetto

IN 3.2V A 30V
LT3514
T1 FQB34N20L
R2 100k
R3 10ohm
R1 1k
D1
C4 100n
C12 2.2µ
D3 BZT52C6V8-7-F
D2 BZT52C36-7-F
L1 10µH
C3 22µ
C2 4.7u
PG
C5 100n
R4 18.2k
SKY SW5 VIN VIN EN RUN RUN RUN RT GND
SW1 DA1 FB1 SW3 DA3 FB3 SW4 DA4 FB4
L4 4.2µH
D6
C10 82p
R9 22k
R10 10k
C11 22µ
2.5V/1A
L3 4.7µH
D5
C8 47p
R7 33k
R8 10k
C9 47µ
3.3V/2A
L2 8.2µH
D4
C6 22p
R5 53.6k
R6 10k
C7 22µ
5V/1A

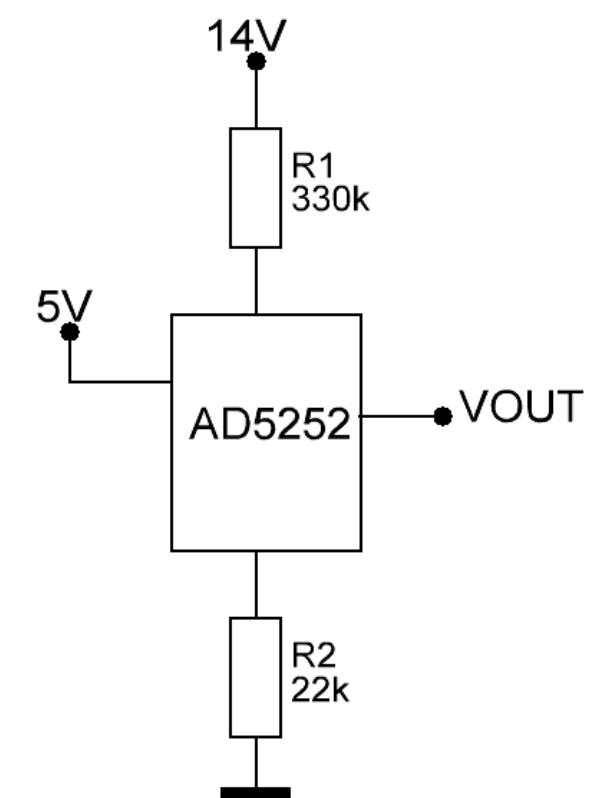
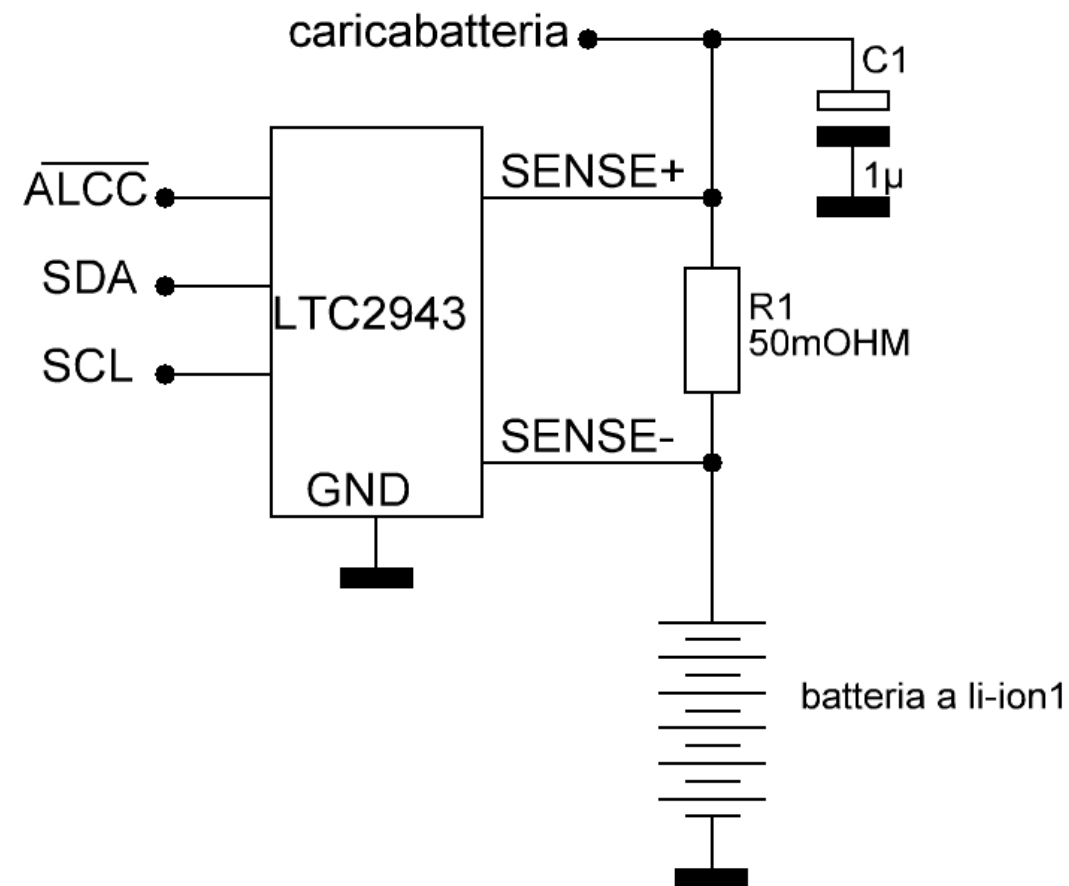
ALIMENTATORE TRIPLO

Nello schema in figura è riportato un alimentatore switching con tre uscite singole. Questo alimentatore è ottenuto utilizzando un modulo linear denominato "LT3514". Questo semplicissimo circuito , visto l'alto amperaggio che riesce ad erogare, può essere impiegato in diverse applicazioni. L'LT3514 dispone anche di un'uscita per il monitoraggio della tensione.

Idee di progetto

AMPLIFICATORE MONO CLASSE DG

In figura è riportato lo schema di un amplificatore di classe DG della Maxim con un potenza massima di 3.3W. Questi sono amplificatori a commutazione utilizzati spesso per sorgenti digitali la cui massima efficienza li rende particolarmente utili nel campo audio.

POTENZIOMETRO REGOLABILE CON AD5252

In questa idea di progetto è stato proposto lo schema circuitale di un trimmer digitale della Analog Device "AD5252". Questo integrato al suo interno incorpora due potenziometri digitale con una resistenza massima di 10Kohm.

Idee di progetto

RILEVATORE DI CARICA

Nella figura seguente è riportato un rilevatore di carica ,ideale per applicazioni in cui serve un monitoraggio costante della carica di una batteria . Il circuito è realizzato con un integrato della Linear "LTC2943" , il quale presenta il vantaggio di poter monitorare costantemente la sua temperatura ,per evitare la rottura dello stesso integrato.

SCAMBIO DATI VIA BLUETOOTH

Ecco come implementare un "cable replacement" via Bluetooth in modo semplice

Tra i sistemi di comunicazione dati wireless, i dispositivi Bluetooth presentano molte caratteristiche interessanti: offrono una discreta velocità, una buona affidabilità e sicurezza nella comunicazione, ed inoltre sono estremamente diffusi (circa un miliardo di dispositivi) e disponibili in molti tipi di apparati dai PC ai PDA fino ai telefoni cellulari. Tutte queste caratteristiche lo rendono adatto in numerose applicazioni, in particolare di tipo industriale, di remotizzazione sensori o di cable replacement di applicazioni legacy. In questo articolo, dopo una breve esposizione delle caratteristiche dello standard, saranno descritti alcuni moduli di interfaccia Bluetooth adatti ad essere utilizzati nei sistemi embedded. Quindi, vedremo una applicazione di cable replacement di esempio in cui interfacceremo un modulo bluetooth con un microcontrollore PIC18F2550 per controllare degli I/O. L'applicazione potrà essere usata con un qualsiasi programma di comunicazione da terminale su PC o PDA/Smartphone, o perfino su un telefono cellulare, se provvisto di un programma adatto (ad esempio sviluppato in Java).

LO STANDARD BLUETOOTH

Bluetooth è una tecnologia di comunicazione wireless a corto raggio nata con l'obiettivo di garantire l'interoperabilità tra dispositivi di tipo molto diverso tra loro - PC, PDA, telefoni cellulari, periferiche - nonché di produttori diversi, richiedendo inoltre bassi consumi rispetto ad altre tecnologie wireless. I dispositivi Bluetooth attualmente in commercio supportano le versioni 1.1, 1.2 o 2.0 EDR delle specifiche Bluetooth. Le versioni 1.x offrono velocità di trasferimento massime di 723 Kbps, mentre la versione 2.0 EDR prevede una velocità di base di 1Mbps, velocità che sale a 2/3 Mbps nei dispositivi EDR (*Enhanced Data Rate*). Sono disponibili 3 classi di potenza RF, ma la stragrande maggioranza dei dispositivi commerciali appartiene a sole due classi, la 1 e la 2. La potenza RF in antenna massima deve essere di 20 dBm (100 mW) per i dispositivi di classe 1, di 4 dBm (2.5 mW) per quelli di classe 2. Le distanze teoriche in spazio aperto sono di 100 m per i dispositivi di classe 1 e di 30 m per quelli di classe 2: tali distanze possono essere incrementate tramite l'uso di antenne esterne di vario tipo. I dispositivi Bluetooth operano nella banda ISM (*Industrial Scientific Medical*) dei 2.4 GHz: il sistema RF utilizza una portante FHSS (*Frequency Hopping Spread Spectrum*) la cui

Figura 1: topologia delle reti piconet e scatternet

frequenza varia continuamente tra i 79 canali disponibili in modo da diminuire le interferenze ed i fenomeni di fading. Ciò permette ad esempio di rendere possibile la coesistenza (almeno in via teorica) tra dispositivi Bluetooth e Wi-Fi. La tecnica AFH (*Adaptive Frequency Hopping*), introdotta a partire dalla versione 1.2, ha portato ad un ulteriore miglioramento in tal senso. Un canale RF è condiviso in un dato istante da un gruppo di dispositivi Bluetooth sincronizzati da un clock comune e da uno schema di hopping (salti di frequenza): il dispositivo che fornisce il riferimento è detto *master*, mentre gli altri sono detti *slave*. Tale insieme di dispositivi forma una *piconet*: essa comprende un master ed un massimo di 7 slave attivi. I dispositivi così sincronizzati cambiano la frequenza della portante fino a 1600 volte al secondo: ciò permette di diminuire la probabilità di intercettazioni delle comunicazioni da parte di dispositivi non autorizzati.

Topologia delle reti Bluetooth

La topologia di base di una rete di dispositivi Bluetooth è, come visto, una piconet: in essa il dispositivo master comunica con gli slave mediante connessioni point-to-point o point-to-multipoint; gli slave non comunicano fra loro direttamente ma tramite il master.
Un dispositivo Bluetooth può essere coinvolto contemporaneamente in più di una piconet sulla base di una suddivisione temporale: in questo caso la piconet è detta scatternet. Notiamo che un dispositivo può essere master di una sola piconet. In figura 1 sono raffigurati due esempi di topologie piconet e scatternet. Bluetooth è una tecnologia di comunicazione wireless ad-hoc: è necessario perciò un insieme di procedure operative atte a formare una piconet in maniera da rendere possibile la comunicazione tra i dispositivi. Le procedure principali sono due:

- Discovering procedure
- Paging procedure

La discovering procedure viene utilizzata da un dispositivo Bluetooth per rilevare altri dispositivi (purchè questi siano disponibili ad essere rilevati: discoverable). La paging procedure viene usata da un dispositivo Bluetooth per stabilire una connessione con un altro specifico dispositivo, purchè quest'ultimo sia nello stato di *page scanning*. Ultimata la paging procedure, un dispositivo può trovarsi in diversi modi operativi tra cui:

- Connected mode
- Parked state

Nel connected mode, i dispositivi sono attivi e connessi fra loro così da poter comunicare. Un dispositivo può uscire da questo modo o sconnettendosi dal canale fisico o entrando nel parked state. Nel parked state il dispositivo rimane connesso alla piconet, ma non può comunicare con gli altri dispositivi: può, però ritornare attivo mediante la procedura di *unparking* invocata dal master. In tal modo è possibile realizzare piconet con più di 7 slave che vengono attivati quando necessario senza dover ripetere la procedura di connessione.

La sicurezza e la procedura di pairing

Il protocollo Bluetooth fornisce diversi strumenti per garantire la sicurezza delle comunicazioni. Ogni dispositivo Bluetooth è innanzi tutto fornito di un indirizzo unico che lo identifica, denominato BD_ADDR (*Bluetooth Device Address*): un numero di 48 bit fornito sotto la supervisione dell'IEEE, simile al MAC dei dispositivi Ethernet. Per permettere la connessione fra due dispositivi Bluetooth è sempre necessario (almeno all'atto della prima connessione) effettuare la procedura di *pairing*: in essa i dispositivi utilizzano un numero PIN (*Personal Identification Number*), ricavato da una stringa scelta dall'utente (*passkey*) o memorizzata nel firmware del dispositivo nel caso di sistemi privi di interfaccia utente. Sulla base del PIN, del BD_ADDR dell'altro dispositivo e di un numero casuale, viene innescata una procedura al termine della quale viene generata un *link key* che sarà memorizzata dai due dispositivi in modo da autenticarli. In aggiunta è possibile utilizzare una connessione crittografata mediante una procedura che genera una *encription key* sulla base di una link key preesistente. Vedremo, quando tratteremo l'applicazione d'esempio, come effettuare in pratica la procedura di pairing.

Figura 2: Stack del protocollo Bluetooth

I layer dello stack Bluetooth

Lo stack Bluetooth è piuttosto complesso: in figura 2 ne possiamo vedere una rappresentazione semplificata in cui sono raffigurati solo i livelli ed i servizi di maggior interesse per i nostri scopi. Possiamo innanzitutto distinguere i livelli tra:

- **Controller layers:** sono i livelli implementati nell'interfaccia Bluetooth vera e propria, ad esempio un adattatore USB.
- **Host layers:** i livelli implementati nel dispositivo host, come un PC o un PDA. Vedremo però che, negli adattatori Bluetooth per sistemi embedded, alcuni, se non tutti i layer host, saranno implementati direttamente negli stessi, sia per semplificare lo sviluppo di applicazioni che per sopperire alla scarsità di risorse hardware dei dispositivi host. Esaminiamo ora brevemente i compiti dei vari layer con i principali protocolli associati. I controller layers sono i seguenti:
- **Radio Frequency:** si occupa della trasmissione e ricezione dei pacchetti sul canale fisico RF.
- **Baseband:** si occupa della gestione dell'accesso al mezzo di trasmissione. Utilizza il *link control protocol* (LC) per effettuare il controllo di flusso e la ritrasmissione dei dati in caso di errore.
- **Link Manager:** oltre che della creazione e gestione dei link logici (mediante i quali sono trasportati i dati), si occupa delle operazioni di discovery di altri dispositivi Bluetooth, o di connessione (paging) con gli stessi, nonché di rendere il dispositivo discoverable o disponibile ad accettare connessioni. Da questo livello è inoltre gestito il pairing tra dispositivi. Tra i controller layers e gli host layers troviamo la HCI (*Host Controller Interface*), l'interfaccia tramite la quale un host comunica con i tre layer inferiori che costituiscono il controller. Vi si accede normalmente tramite il driver del dispositivo Bluetooth.

Gli host layers sono:

- **L2CAP:** *Logical Link Control and Adaptation Protocol*. Si occupa della creazione e gestione dei canali (channels) per il trasporto dei dati: utilizza a questo scopo il protocollo L2CAP per stabilire una connessione fra gli endpoint dei dispositivi peer. Può opzionalmente fornire funzionalità di recupero errori e di controllo di flusso aggiuntive a quelle fornite dal livello Baseband.
- **Application:** protocolli SDP, RFCOMM, BNEP, OBEX; profili GAP, PAN, SPP, FTP, ecc. I profili definiscono le possibili applicazioni di un dispositivo Bluetooth: si avvalgono per il loro funzionamento di uno o più protocolli. Possono dipendere anche da altri profili. Vediamo una breve descrizione dei profili e dei protocolli che risultano più interessanti per i nostri scopi. Tra i profili ricordiamo i seguenti:
- **GAP:** Generic Access Profile, il profilo di base che tutti i dispositivi Bluetooth devono supportare per permettere l'interoperabilità fra gli stessi. Permette inoltre di interrogare i dispositivi remoti riguardo alle applicazioni (o servizi) che essi supportano.
- **SPP:** Serial Port Profile. Tale profilo permette di realizzare il cosiddetto cable replacement di connessioni legacy RS-232 cablate. Vedremo che è il profilo supportato praticamente da tutti moduli Bluetooth per sistemi embedded. Le specifiche di questo profilo prevedono velocità massime di 128 Kbps, ma spesso le implementazioni superano tale limite. Tale profilo sfrutta RFCOMM per emulare la porta seriale (vedi oltre).
- **DUN:** Dial-Up Networking, è probabilmente il profilo più noto poiché permette di connettersi ad una rete WAN mediante dial-up. Lo scenario più comune è quello di un telefono cellulare Bluetooth usato per connettere un PC ad Internet mediante GPRS/UMTS/HSDPA. È basato sul profilo SPP.
- **PAN:** Personal Area Networking, permette di creare una rete ad-hoc tra due o più dispositivi Bluetooth, ed anche di ottenere l'accesso ad una rete LAN o WAN mediante un Access Point Bluetooth.

Tra i protocolli del layer inferiore vi sono:

- **SDP:** *Service Discovery Protocol*, il protocollo mediante il quale un SDP client può ricevere informazioni dall'SDP server di un dispositivo remoto riguardo i servizi (ad esempio SPP, PAN o altro) che è in grado di fornire.
- **OBEX:** *OBject EXchange*, un protocollo di comunicazione client-server che permette lo scambio di oggetti dati tra dispositivi. È usato da diversi profili tra cui FTP (File Transfer Profile) ed OPP (Object Push Profile).
- **RFCOMM:** *Radio Frequency COMMunication*. Questo protocollo emula il funzionamento di una porta seriale RS-232 su un canale Bluetooth. È in grado di emulare anche tutti i segnali di controllo RS-232 e crea automaticamente una connessione null-modem quando i dispositivi Bluetooth sono entrambi DTE (ad esempio due PC). È utilizzato da numerosi profili tra cui SPP, DUN, e dal protocollo OBEX.
- **BNEP:** *Bluetooth Network Encapsulation Protocol*; permette di incapsulare pacchetti provenienti da protocolli di networking come IPv4 e IPv6. Viene usato dal profilo PAN. Questa lunga introduzione sulle caratteristiche dello standard Bluetooth è stata necessaria per acquisire una certa padronanza dei concetti necessari per comprendere le caratteristiche e l'uso dei moduli Bluetooth che vedremo nel seguito.

MODULI ADATTATORI BLUETOOTH PER SISTEMI EMBEDDED

Esamineremo ora alcuni moduli adattatori Bluetooth per sistemi embedded: i moduli Parani ESD100 ed ESD200 di Sena Technologies (www.sena.com). Nella tabella in figura 3 possiamo vedere un raffronto delle caratteristiche dei moduli. I moduli in questione sono adatti per ottenere il cable replacement di applicazioni scritte per fare uso dell'interfaccia RS-232: implementano infatti il profilo SPP tramite il quale è possibile emulare la porta RS-232 in maniera trasparente per l'applicazione. Sono inoltre dotati di antenna integrata. Questi adattatori implementano sia i controller layer che parte degli host layer: In particolare sono sempre implementati, tramite un microcontrollore integrato, i layer (ed il protocollo relativo) L2CAP, il protocollo RFCOMM ed il profilo SPP. Verso l'host si collegano tramite una interfaccia UART. Le velocità massime ottenibili si aggirano sui

Modello	Parani ESD 100	Parani ESD200
Produttore	Sena Technologies www.sena.com	Sena Technologies www.sena.com
Specifica Bluetooth	v. 1.2	v. 1.2
Classe	1; Pmax 18 dBm (63 mW)	2; Pmax 4 dBm (2.5 mW)
Protocolli host	RFCOMM SDP L2CAP	RFCOMM SDP L2CAP
Profili	GAP, SPP	GAP, SPP
Interfaccia Host	UART (max 230400 Baud)	UART (max 230400 Baud)
Throughput massimo	230 Kbps	230 Kbps
Altre interfacce	-	-

Figura 3: Caratteristiche dei moduli Bluetooth esaminati

COMANDO	DESCRIZIONE
AT+UARTCONFIG, baudrate, parità, Stopbit, HwFc	Setta i parametri dell'UART: - Baud rate: 1200-230400 baud - Parità N/E/O - bit di stop: 1 o 2 - Hardware flow control: no= 0/sì=1
AT+BTKEY="<stringa>"	Setta il PIN code: è una stringa alfanumerica di max 16 caratteri
AT+BTSEC, Authentication, Encryption	Abilita l'autenticazione (tramite PIN code) ed in aggiunta la crittografia
AT+BTMODE, n	Setta la modalità operativa del modulo: - Mode0: attesa di comandi AT - Mode1: tenta di connettersi all'ultimo dispositivo Bluetooth già connesso in precedenza (modalità master) - Mode2: attende una connessione dall'ultimo dispositivo Bluetooth già connesso in precedenza (modalità slave) - Mode3: attende una connessione da parte di un qualsiasi dispositivo Bluetooth (è discoverable ed in page scan)
ATSnn=mm	Memorizza il valore mm del registro nn
ATZ	Effettua un reset Software

Figura 4: Comandi AT del modulo Bluetooth Parani usati nell'applicazione d'esempio

230 Kbps per un collegamento half-duplex, diminuiscono sui 170 Kbps per un collegamento fullduplex. Nello standard Bluetooth infatti, i collegamenti fullduplex vengono simulati intercalando i pacchetti che vengono trasmessi dal master con quelli trasmessi dallo slave. Come si vede siamo lontani dalla velocità massima di 723 Kbps degli standard Bluetooth 1.x: ciò è dovuto in parte a delle elaborazioni effettuate dallo stack per emulare la connessione RS-232 sul canale RF, in parte alle risorse limitate dei moduli. Comunque le velocità ottenibili sono più che adeguate per gli scopi di cable replacement. I moduli Parani presentano un connettore a passo standard da 2.54 mm.

CABLE REPLACEMENT

Per l'applicazione d'esempio ho scelto di utilizzare un modulo Parani ESD200: sebbene sia meno dotato del modulo Free2Move, è più semplice da usare e comunque, dato che verrà interfacciato con un microcontrollore, quest'ultimo potrà fornire gli I/O che ci servono. Questo è ciò di cui abbiamo bisogno per la nostra applicazione di cable replacement. Come già detto, il modulo ESD200 si interfaccia all'host tramite UART: il modulo viene configurato e gestito mediante un set di comandi AT esteso. È fornito un insieme di 31 comandi AT suddivisi nelle seguenti categorie:

- *Reset:* reset software o hardware del modulo
- *Serial Port:* configurazione UART
- *Bluetooth:* informazioni sulla configurazione, ricerca dispositivi, connessione, gestione della sicurezza
- *S-Register:* lettura/scrittura di registri contenenti altri parametri di configurazione del modulo

Nella figura 4 sono descritti i comandi AT usati nell'applicazione.

L'hardware dell'applicazione

L'hardware si basa sul microcontrollore PIC18F2550, appartenente alla famiglia di micro ad 8 bit avanzati PIC18. Questi sono dei processori RISC con architettura Harvard modificata: possiedono un set di 75 istruzioni rispetto alle 35 della famiglia PIC16, sono inoltre ottimizzati per l'uso di un compilatore C. Un set esteso di 8 istruzioni aggiuntive può essere usato per ottimizzare ulteriormente l'uso del compilatore. Il PIC18F2550 in particolare possiede 32 KB di memoria Flash (16 K x 16), 2 KB di RAM e 256 B di EEPROM. Comprende un moltiplicatore hardware 8 x 8 bit ed ha una dotazione molto ricca di periferiche, tra cui un USART ed anche una interfaccia USB Full Speed. Può funzionare alla velocità di clock massima di 48 MHz. Nella descrizione dell'hardware che segue facciamo riferimento allo schema elettrico in figura 5. Il pin 8 (RXD – ingresso dati UART) del modulo Parani ESD200 è collegato al pin 17 che corrisponde al TX dell'USART del PIC, mentre il pin 7 (TXD – uscita dati UART) è collegato al pin 18 (RX) del PIC. I

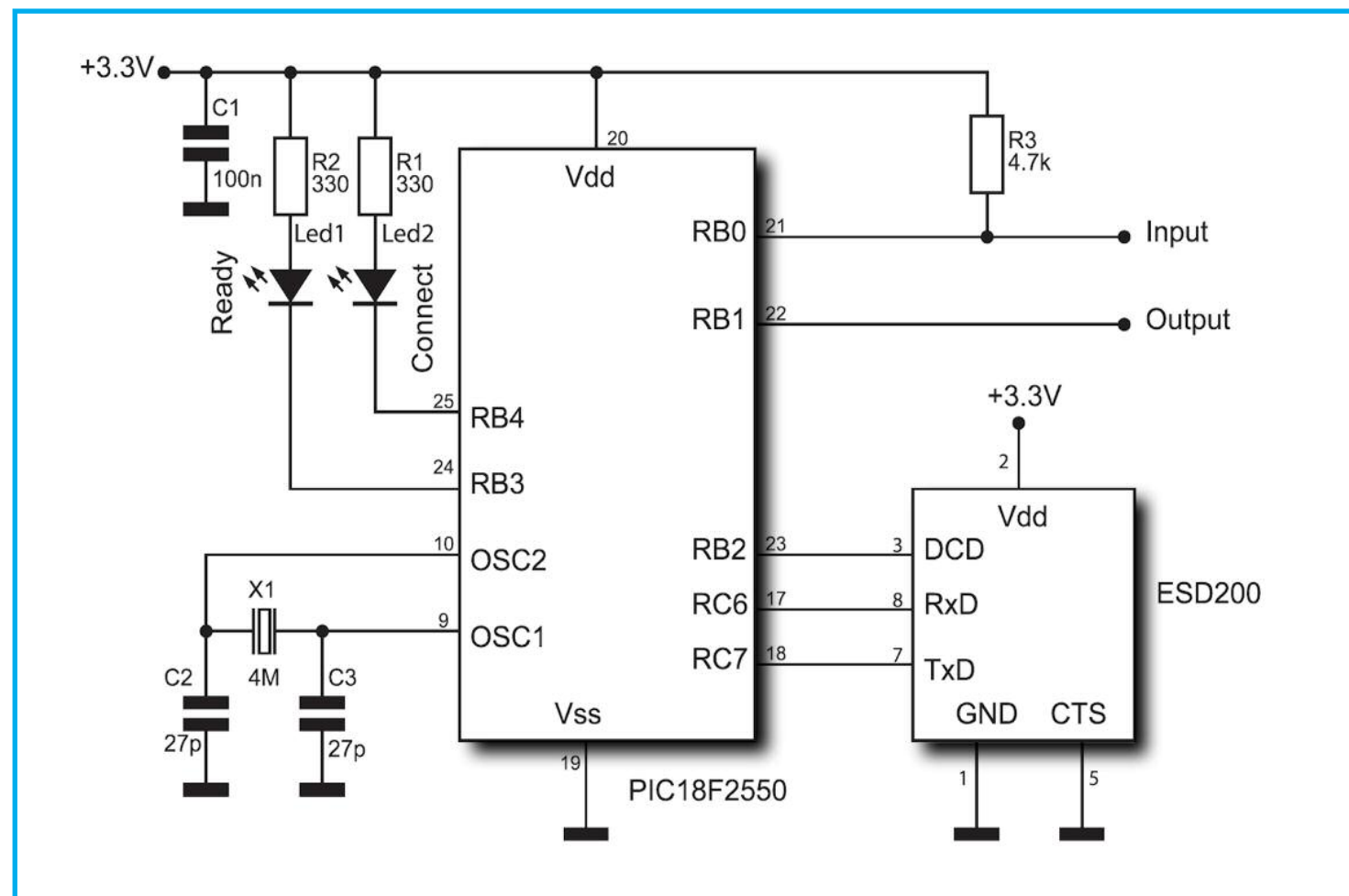



Figura 5: Schema elettrico del sistema di comunicazione Bluetooth

moduli Parani possono essere usati con o senza controllo di flusso hardware: poiché nella nostra applicazione non lo useremo, è necessario porre a massa il pin 5 (CTS - Clear To Send). Il controllo di flusso hardware è consigliato quando si usa un baud rate dell'UART superiore a 38400 baud: volendo è relativamente semplice implementare tale controllo dedicando due pin del PIC18 ai segnali CTS ed RTS (Ready To Send). Il pin 3 (DCD – Bluetooth Connect Detect) dell'ESD200 è collegato al pin 23 (RB2) del PIC per segnalare l'avvenuta connessione del modulo Bluetooth: questo sarà segnalato dall'accensione del LED2. Il LED1 viene usato per segnalare la condizione di 'Ready' del modulo, come vedremo quando parleremo del firmware del PIC. Il clock del sistema viene generato tramite il quarzo collegato tra i pin 9 e 10 (OSC1 e OSC2) del PIC. Il PIC è configurato per l'uso di un PLL interno che genererà a partire dai 4 MHz del quarzo il segnale di clock reale del micro a 24 MHz. Per interfacciare il PIC col mondo esterno, useremo due pin di I/O: il pin 21 (RB0) verrà configurato dal firmware come ingresso TTL, mentre il pin 22 (RB1) verrà configurato come uscita.

Le uscite del PIC possono sopportare un carico massimo di 25 mA e sono compatibili CMOS/TTL. Volendo, potremo naturalmente utilizzare dei relè o dei fototriac per poter pilotare dei carichi diversi, e usare degli optoisolatori per gli ingressi. Il tutto verrà alimentato tramite una tensione stabilizzata di 3.3 V, con una corrente massima di circa 300 mA.

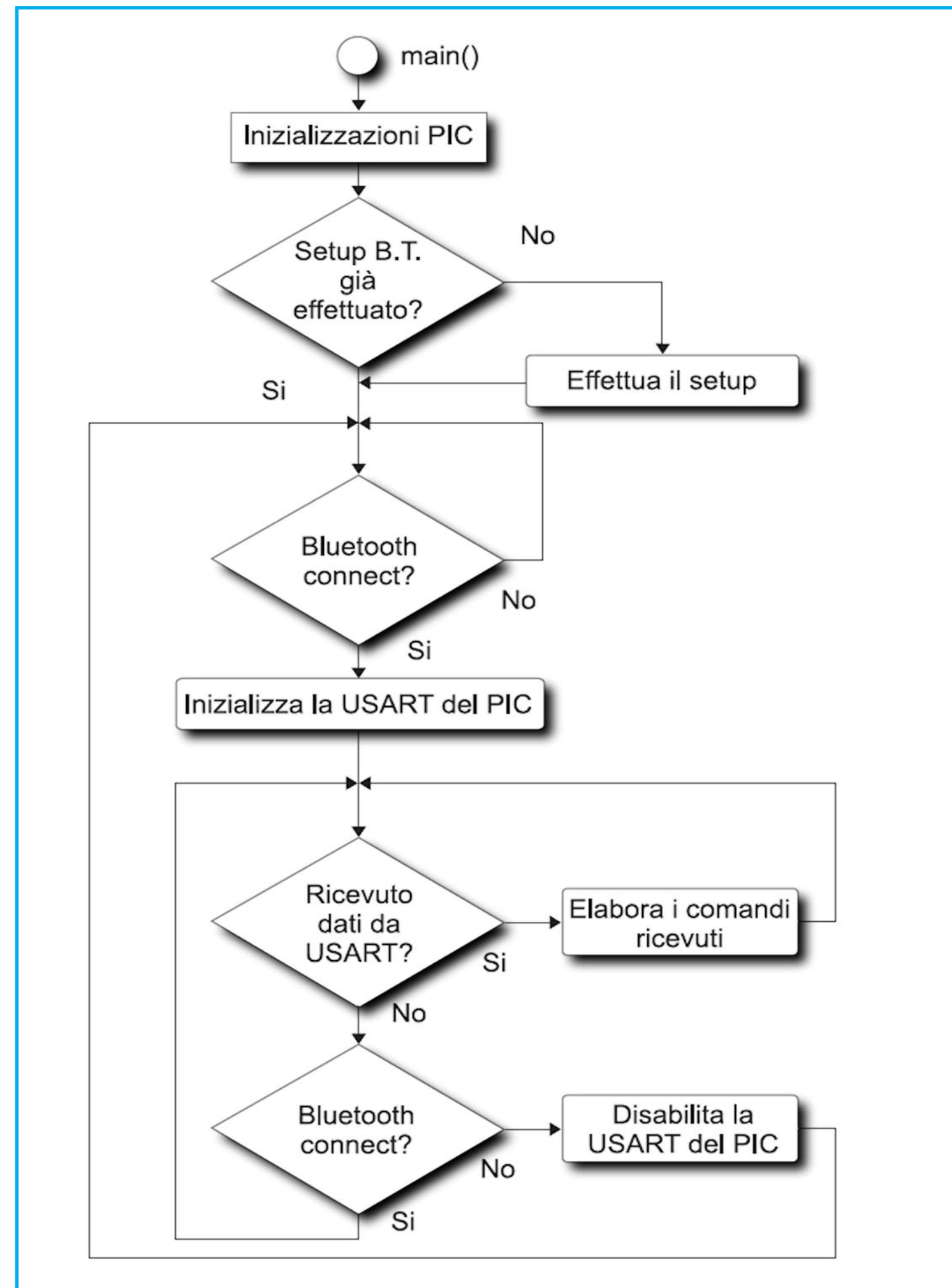



Figura 6: Flow chart della funzione main()

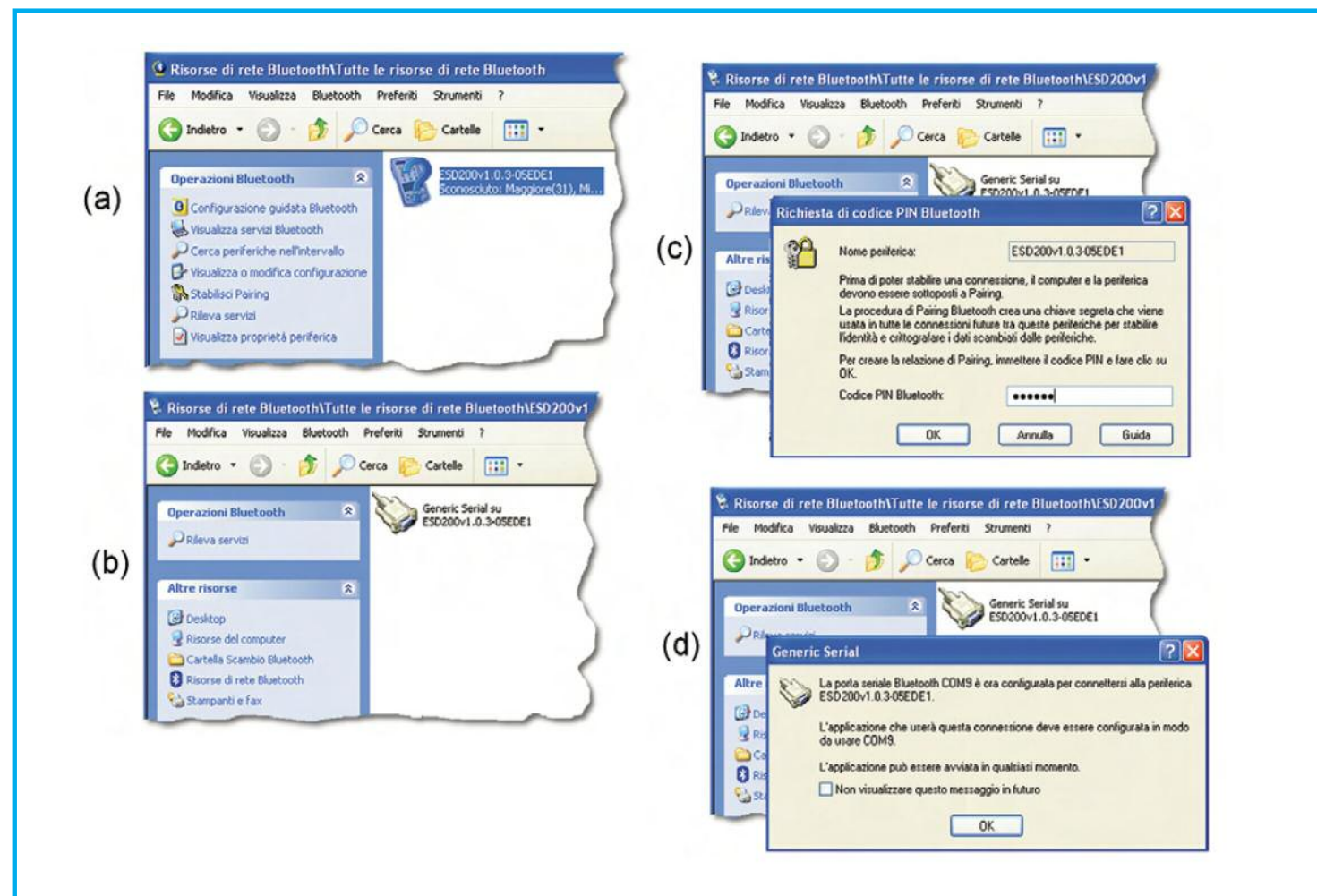



Figura 7: Procedura di pairing del modulo Bluetooth

Il firmware

Il firmware è stato sviluppato tramite l'ambiente di sviluppo MPLAB ed il compilatore C18 di Microchip. La funzione principale *main()* è rappresentata nel flow chart in figura 6. In questa funzione vengono innanzi tutto inizializzate le porte di I/O del PIC, quindi, se necessario, viene effettuato il setup del modulo Bluetooth richiamando la funzione *BTSetup()*. Completato il setup, verrà settato il flag SETUP_DONE nella memoria EEPROM, flag che viene controllato ad ogni accensione del PIC. La funzione *BTSetup()* configura il modulo Bluetooth tramite i comandi AT descritti nella tabella in figura 4: infatti i moduli Parani sono settati in fabbrica in modo da funzionare con un baud rate di 9600 baud e posti in una modalità di funzionamento di attesa di comandi AT (Mode0). La procedura di configurazione si occupa in particolare di settare l'interfaccia UART a 57600 baud, nessuna parità, e senza controllo di flusso, tramite il comando

```
AT+UARTCONFIG,57600,N,1
```

quindi, viene impostato il PIN del dispositivo, ad esempio

```
AT+BTKEY=”123456”
```

si abilita allora l'autenticazione con il comando

```
AT+BTSEC,1,0
```

Il comando seguente pone il modulo Bluetooth nella modalità di funzionamento Mode3 in cui si predispone ad accettare connessioni da altri dispositivi (è cioè discoverable ed in page scan)

```
AT+BTMODE,3
```

Si disabilitano ora i reply da parte del modulo infatti normalmente quest'ultimo segnala il proprio stato con i messaggi “OK”, “ERROR”, “CONNECT” o “DISCONNECT” che interferirebbero con il funzionamento dell'applicazione di wireless UART

```
ATS10=0
```

Infine il comando ATZ effettua un reset software per abilitare la nuova configurazione:

```
ATZ
```

Il completamento della fase di setup viene segnalato dall'accensione del LED1, quindi il processo rimane in attesa di connessione da parte di un altro dispositivo Bluetooth: l'avvenuta connessione viene segnalata dal pin 3 (DCD) del modulo Bluetooth ed all'utente dall'accensione del LED2. Il PIC si predispone allora ad accettare comandi tramite il modulo Bluetooth, utilizzandolo come se fosse un UART in maniera del tutto trasparente: viene innanzitutto abilitata l'USART del PIC settandola a 57600 baud, quindi si entra nel ciclo di polling di ricezione dati.
La funzione *ProcCommand()* si occupa di elaborare il comando ricevuto: se è 'I' il PIC risponde inviando il valore letto sul pin di input RB0, se il comando è O0 od O1 il PIC pone il pin di output RB1 al valore 0 od 1 rispettivamente, quindi invia tale valore come conferma. Se il comando ricevuto è errato, viene inviato il carattere '?'. Se non ci sono dati in attesa di essere letti nell'USART, il PIC controlla che la connessione Bluetooth sia ancora attiva leggendo il segnale DCD, se non lo è disabilita l'USART ed esce dal ciclo di ricezione dati. Rimane quindi in attesa di una nuova connessione e così via. Il codice della funzione *main()* è riportato nel listato 1. Si noti che il firmware sviluppato è perfettamente compatibile con i moduli ESD100, mentre i pin hanno una disposizione differente.

Configurazione del PC ed uso del sistema Bluetooth

Una volta acceso il nostro sistema attendiamo che il setup del modulo Bluetooth sia completato, come segnalato dall'accensione del LED1 'Ready'. La prima volta che vorremo usare la nostra applicazione, sarà innanzi tutto necessario configurare il PC: nel seguito facciamo riferimento ad un PC con sistema operativo Windows XP (la configurazione è analoga se usiamo un PDA con Pocket PC o Windows Mobile). Clicchiamo col tasto destro del mouse sull'icona Bluetooth presente sulla barra delle applicazioni e selezioniamo la voce *Esplora Risorse di Rete Bluetooth*. Dalla schermata relativa clicchiamo sulla voce di menu *Bluetooth>Cerca periferiche*. Nel giro di qualche secondo comparirà all'interno della finestra l'icona del modulo Bluetooth ESD200 (figura 7a). Facciamo doppio click sull'icona (o clicchiamo sulla voce *Rileva servizi*): comparirà una icona a forma di connettore seriale relativa al servizio *Porta Seriale* (figura 7b). È necessario ora effettuare il pairing tra il PC ed il modulo. Ancora doppio click sull'icona del connettore seriale: sulla icona Bluetooth della barra della applicazioni comparirà un fumetto indicante la richiesta di connessione (figura 7c): clicchiamo sul fumetto ed immettiamo il PIN corrispondente a quello memorizzato nel firmware del modulo (nel nostro esempio “123456”). Completata la procedura comparirà una message box che ci in-

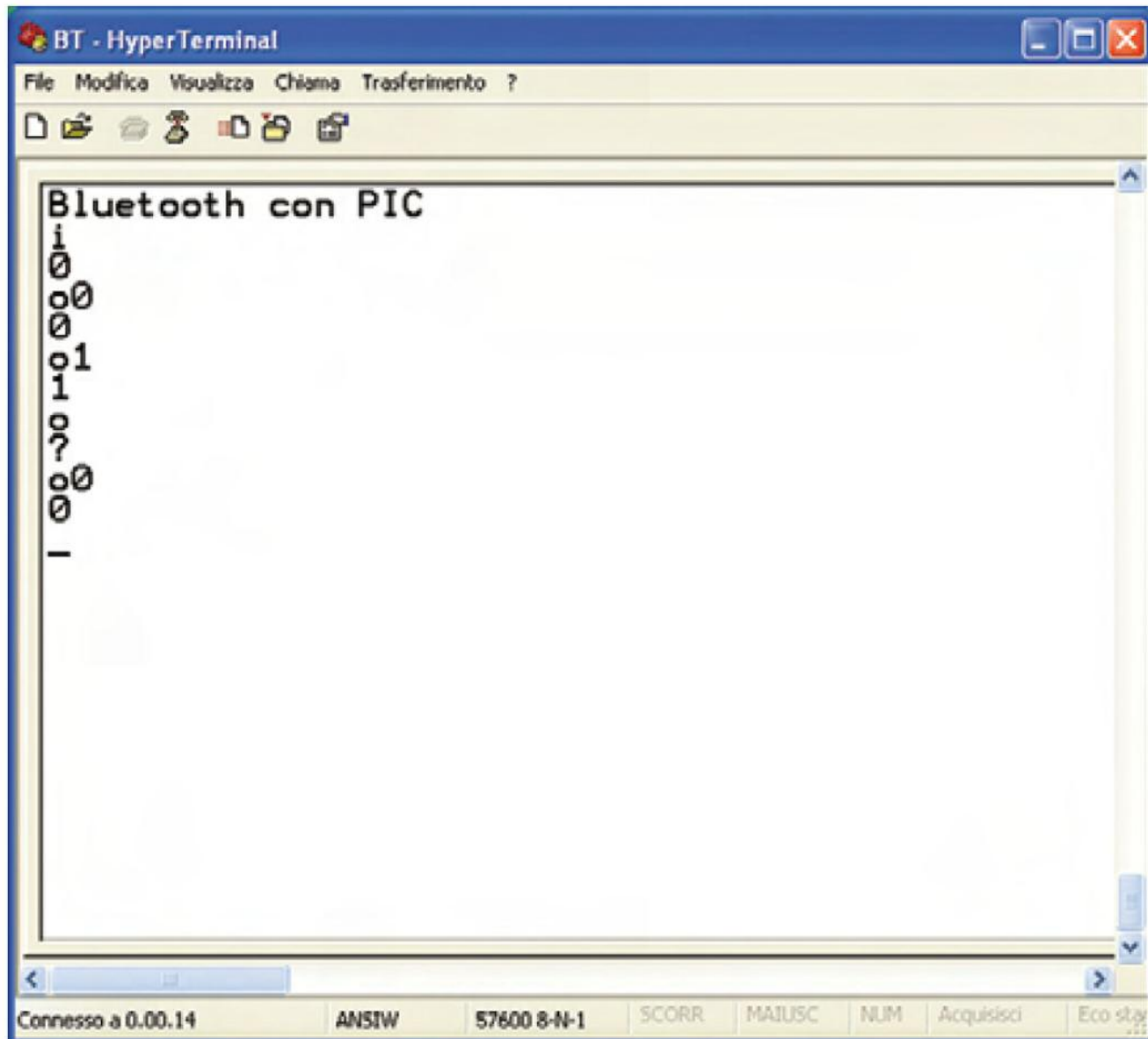



Figura 8: Comunicazione Bluetooth con PIC e Hyperterminal

dica quale porta COM virtuale è necessario usare per comunicare col modulo Bluetooth (figura 7d). Chiudiamo la connessione cliccando col tasto destro del mouse sull'icona della seriale e selezionando la voce *Disconnetti porta seriale Bluetooth*. D'ora in poi possiamo comunicare con il modulo direttamente con una qualsiasi applicazione di terminale, ad esempio, lanciamo Hyperterminal (*Accessori>Comunicazione>Hyperterminal*) e creiamo una nuova connessione: nelle dialog box che seguiranno selezioniamo la porta COM virtuale suddetta, la settiamo a 57600 baud e senza controllo di flusso. Quindi dalla voce di menu File>Proprietà selezioniamo la scheda Impostazioni, clicchiamo su Impostazioni ASCII e spuntiamo la casella Eco dei caratteri digitati localmente. All'uscita da Hyperterminal scegliamo di salvare la sessione. Le volte successive sarà sufficiente lanciare la connessione con Hyperterminal per comunicare col sistema. Vediamo in figura 8 una schermata di Hyperterminal con una sessione di collegamento con il modulo.

LISTATO 1

```
void main(void) {
unsigned char i = 0; // INDICE BUFFER
char UC; // CARATTERE USART
char BT_CONNECT = FALSE;
char SETUP_DONE = FALSE;
/*** INIZIALIZZAZIONI PIC ***/
INTCON2bits.RBPU=0;
PORTA = 0;
PORTB = 0;
PORTC = 0;
TRISA = 0B00000000; // PORTA
TRISB = 0B00000101; // PORTB
TRISC = 0B11000000; // PORTC
// SPEGNE LED1 e LED2
LATBbits.LATB3 = 1; // LED1 'Ready'
LATBbits.LATB4 = 1; // LED2 'Connect'
//*** se e' necessario effettua il setup
// del modulo Bluetooth
// legge il flag SETUP_DONE da EEPROM
EECON1bits.EEPGD = 0;
EEADR = 0;
EECON1bits.RD = 1;
SETUP_DONE = EEDATA;
if (SETUP_DONE!=TRUE) {
BTSetup();
EEWrite(TRUE);
}
LATBbits.LATB3 = 0; // accende LED1
//*** CICLO PRINCIPALE
while (1) {
//* ATTESA CONNESSIONE BLUETOOTH
while (PORTBbits.RB2) {
;
}
LATBbits.LATB4 = 0; // ACCENDE LED 2
BT_CONNECT = TRUE; // CONNESSO
//*** CICLO COMUNICAZIONE DATI
//* setup USART per comunicazione BT:
//57600,8,N,1
OpenUSART( USART_TX_INT_OFF &
USART_RX_INT_OFF &
USART_ASYNCH_MODE &
```

(segue)

```
USART_EIGHT_BIT &
USART_CONT_RX &
USART_BRGH_HIGH,
25 );
i = 0;
// cancella il buffer
memset(buffer,0,BUFFLEN);
putrsUSART(“Bluetooth con PIC\r\n”);
//* loop ricezione, FINCHE’ IL
// MODULO BLUETOOTH E’ CONNESSO
while (BT_CONNECT) {
// elabora i caratteri ricevuti
if (DataRdyUSART()) {
switch (UC = ReadUSART()) {
case 0x0D: // FINE comando
// processa il comando
ProcCommand();
putsUSART(buffer);
// risposta
i = 0;
memset(buffer,0,BUFFLEN);
break;
default:
// CARATTERE COMANDO
buffer[i++] = UC; // lo copia
if (i>BUFFLEN) {
// SE buffer pieno
i = 0;
memset(buffer,0,BUFFLEN);
putrsUSART(“\r\n?\r\n”);
}
break;
}
}
/* CONNECT NON PIU’ ATTIVO? */
if (PORTBbits.RB2) {
LATBbits.LATB4 = 1;
// SPEGNE LED 2
BT_CONNECT = FALSE;
delay(200);
}
} // fine while(BT_CONNECT)
CloseUSART(); // DISABILITA l’USART
} // fine while(1) ciclo principale
} // fine main()
```

INVERTER DA 40 WATT

In queste pagine vedremo come realizzare un circuito in grado di erogare una tensione di 220Vac a 50Hz a partire da una tensione continua di 12V ricavata, ad esempio, da una comunissima batteria per auto

Il trasformatore è una macchina elettrica costituita da due avvolgimenti realizzati su uno stesso nucleo ferromagnetico (materiale fortemente permeabile al campo magnetico) come schematizzato in figura 2. Qualsiasi conduttore percorso da corrente genera un campo magnetico e, se tale corrente è alternata, anche il campo magnetico che si genera risulta alternato. Se poi il conduttore è avvolto su se stesso in n spire (solenoide) il campo magnetico generato sarà massimo all'interno dell'avvolgimento e nullo (o comunque trascurabile) all'esterno.
Realizzando l'avvolgimento su un materiale ferromagnetico, il campo magnetico generato darà vita ad un flusso (F) vincolato all'interno del materiale stesso. Il fenomeno è del tutto reversibile, ovvero immergendo un solenoide in un flusso magnetico alternato, si potrà misurare ai suoi capi una tensione indotta (forza magnetomotrice). Realizzando i due avvolgimenti su uno stesso nucleo sarà possibile generare il flusso magnetico alimentando un avvolgimento e prelevare la forza magnetomotrice indotta sull'altro avvolgimento. Alimentando con una tensione sinusoidale di ampiezza V1 l'avvolgimento di n1 spire, la tensione V2 ottenuta sull'altro avvolgimento (di n2 spire) è data da:

$$V2=V1*n2/n1$$

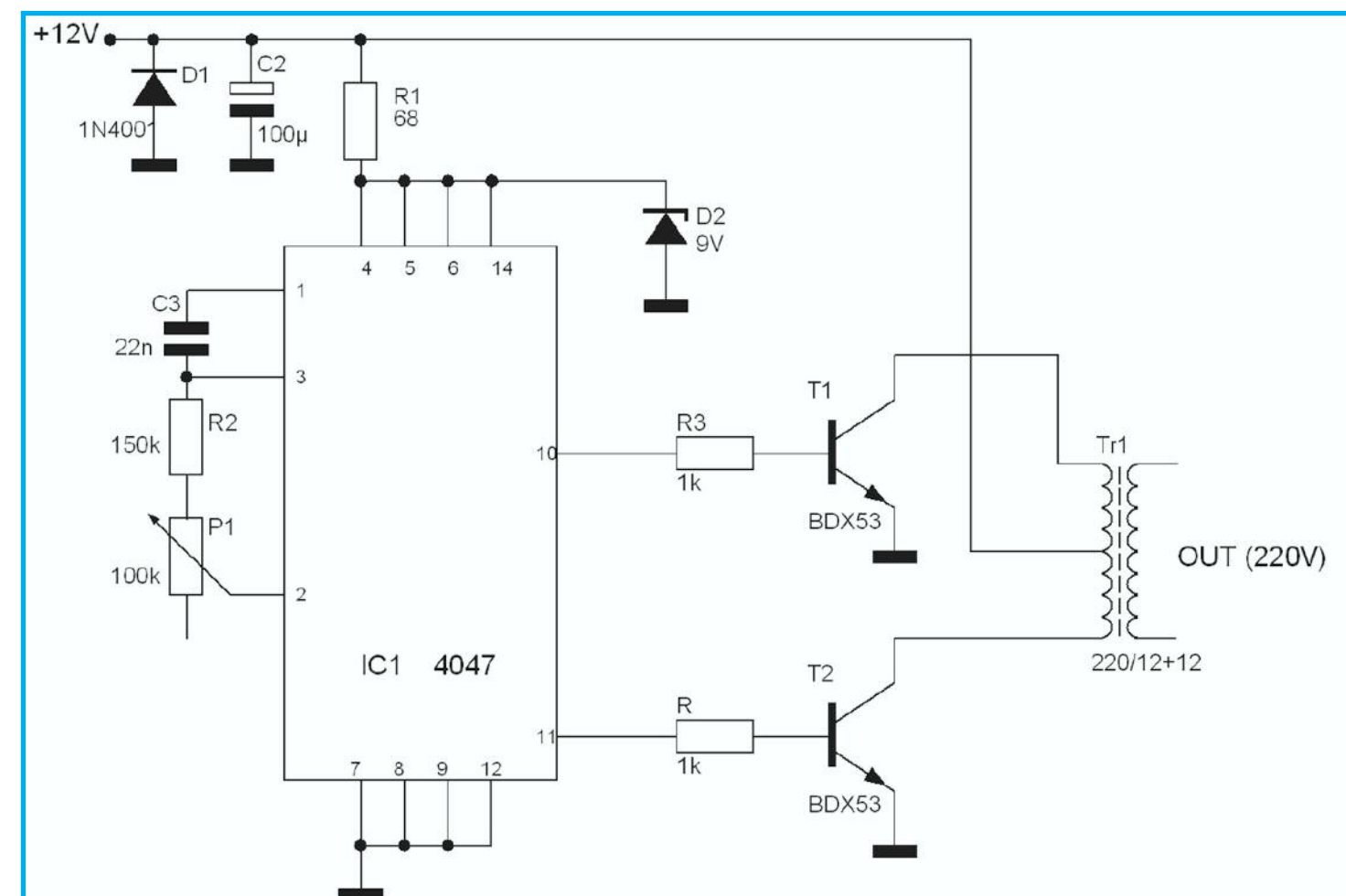



Figura 1 Schema elettrico dell'inverter

Agendo quindi sul numero di spire dei due avvolgimenti, è possibile determinare se la tensione V2 sarà maggiore o minore della tensione V1. L'avvolgimento alimentato è detto avvolgimento primario, mentre l'altro è detto avvolgimento secondario. È opportuno sottolineare che il trasformatore, benché consenta di ottenere, in uscita, tensioni maggiori di quelle in ingresso, non è assolutamente un amplificatore: la potenza all'avvolgimento primario è infatti uguale a quella sul secondario, il che significa che il prodotto tra tensione e corrente è costante sui due avvolgimenti. Se la tensione al secondario è maggiore di quella al primario, sicuramente non sarà lo stesso per la corrente. Disponendo ad esempio di un trasformatore 220V/12V l'avvolgimento a 220V sarà sicuramente realizzato con un filo di diametro minore rispetto all'altro in quanto, a parità di potenza, ad una tensione maggiore deve corrispondere una corrente minore. Gli avvolgimenti sono realizzati con filo di rame smaltato in modo da avere alti numeri di spire in spazi molto ristretti.

L'INVERTER

L'inverter è un circuito in grado di erogare una tensione alternata di 220V per pilotare un carico di rete da 40W a partire da una tensione continua di 12V. Un circuito di questo tipo può essere utile ad esempio per disporre in auto di una presa a 220V come quelle presenti in casa. La figura 1 riporta lo schema elettrico dell'inverter da 40W. Il circuito si basa su un multivibratore astabile che genera due onde quadre con cui si pilotano i transistori T1 e T2. Il multivibratore è il circuito integrato IC1

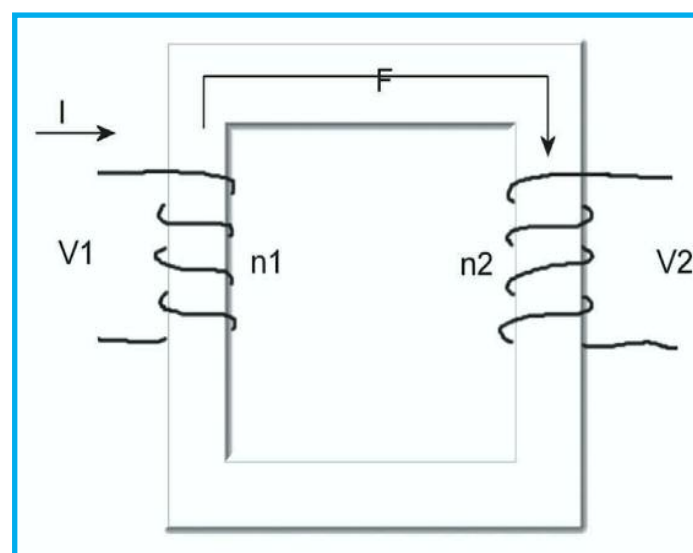



Figura 2 Schema della struttura di un trasformatore

(CD4047) che può funzionare come monostabile o come astabile. Nel nostro caso, collegando i pin 4, 5, 6 e 14 a livello alto ed i pin 7, 8, 9 e 12 a massa, è utilizzato come astabile e alle uscite 10 ed 11 si hanno due onde quadre opposte il cui periodo è dato dalla relazione:

T=4.4 C3*(R2+P1)

dove, con P1 si intende ovviamente la porzione di resistenza inserita mediante il cursore del trimmer P1. Se P1 è tutto inserito (il cursore nella posizione più in basso) il periodo sarà di 24,2msec (corrispondente ad una frequenza di 41,3Hz), mentre con P1 totalmente escluso il periodo è di 14,52msec corrispondente ad una frequenza di 68,8Hz. Agendo su P1 si può dunque regolare la frequenza dell'onda quadra precisamente su 50Hz che è il valore di frequenza della tensione di rete. Le onde quadre così ottenute vengono applicate alla base dei due transistori T1 e T2 i quali si comportano come degli interruttori comandati: quando l'onda quadra è a livello alto (9V) il relativo transistor si comporta come un interruttore chiuso e porta a massa il terminale del trasformatore al quale è connesso; viceversa quando l'onda quadra è a livello basso, il transistor è interdetto ed il relativo avvolgimento del trasformatore non risulta alimentato. Pilotando il trasformatore in questo modo, si ottiene all'uscita un'onda sinusoidale di 220V e frequenza pari alla frequenza dell'onda quadra (quindi 50Hz se si è eseguita una buona taratura). La resistenza R1 consente di fissare a circa 45mA la corrente nel diodo zener D2 il quale stabilizza a 9V la tensione di alimentazione di IC1. C1 filtra verso massa eventuali disturbi in radiofrequenza. Il diodo D1 protegge il circuito contro le tensioni negative, mentre le resistenze R ed R3 limitano la corrente di base nei transistori ad un massimo di circa 8mA garantendo la saturazione dei transitori in caso di livello alto in ingresso. T1 e T2 necessitano di un dissipatore e, poiché il circuito genera una tensione di 220V, si consiglia di prestare particolare attenzione durante il montaggio e la taratura.

LA PRESA CENTRALE

Il trasformatore utilizzato nello schema di figura 1 è dotato di una presa centrale su uno dei due avvolgimenti. Le due metà dell'avvolgimento dotato di presa centrale, sono composte da spire avvolte in senso contrario nelle due parti dell'avvolgimento. Questa particolarità permette di ottenere due tensioni sinusoidali della stessa ampiezza ma di fase opposta (quando la prima è massima positiva, la seconda è massima negativa e viceversa). In figura 3 un particolare dell'avvolgimento con presa centrale. Nel caso di un trasformatore a presa centrale con primario da 220V e secondario da 12V+12V è possibile ottenere due tensioni sinusoidali di ampiezza 12V (ciascuna ai capi di un singolo avvolgimento secondario) oppure una tensione sinusoidale di ampiezza 24V (se si preleva la tensione si due capi opposti dei due avvolgimenti escludendo la presa centrale). Come già spiegato in precedenza, il trasformatore Tr1 dello schema di figura 1, viene pilotato attraverso l'avvolgimento secondario mediante un'onda quadra. Per quanto detto, un trasformatore funziona solo con tensioni sinusoidali, quindi come può lavorare correttamente con un segnale rettangolare? In elettronica esiste un noto teorema (teorema di Fourier) secondo il quale ciascun segnale periodico (sia esso rettangolare, triangolare o di qualsiasi altra forma) avente una certa frequenza fo può essere sempre scomposto nella somma di

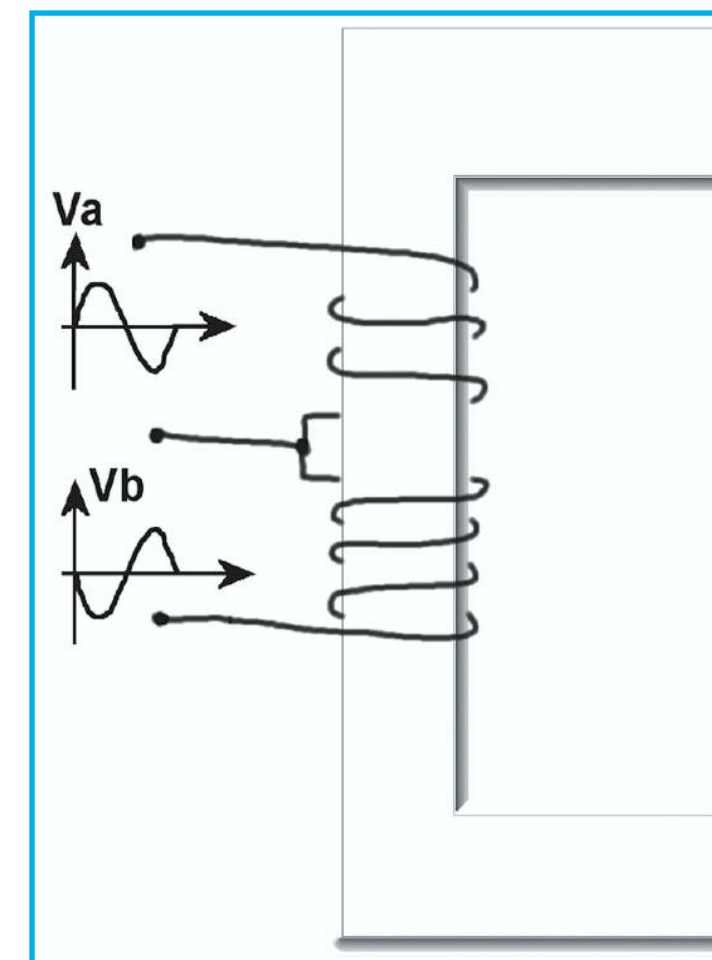



Figura 3 L'avvolgimento con presa centrale

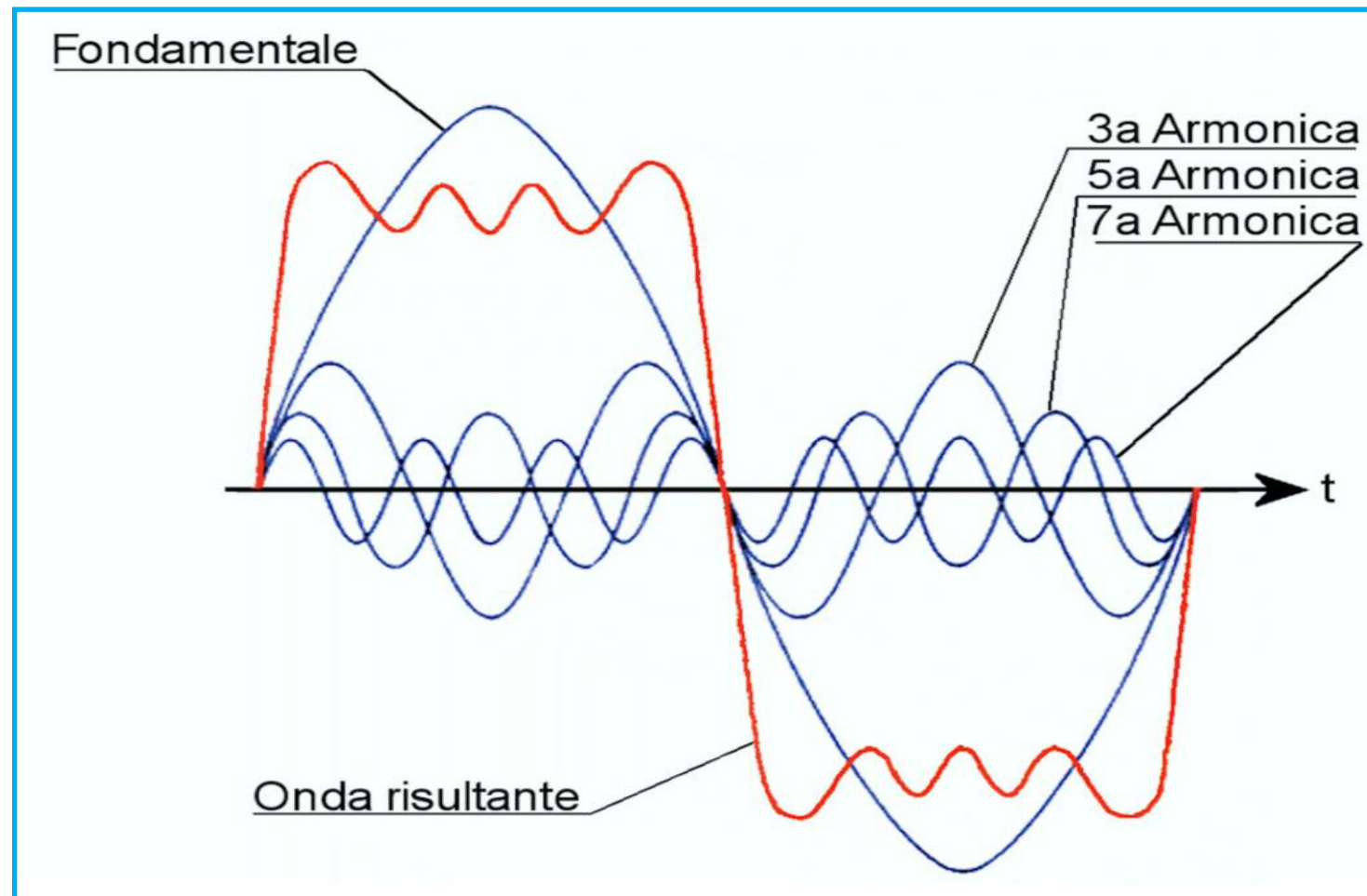



Figura 4 La somma della fondamentale e delle prime tre armoniche di un'onda quadra

un numero infinito di onde sinusoidali di frequenza pari e multipla della frequenza fo ed ampiezza opportuna e via via decrescente. In figura 4 è riportato un esempio: l'onda in rosso è il risultato della somma delle prime tre armoniche non nulle (in blu) e della fondamentale. Si noti che le armoniche di ordine superiore hanno ampiezza minore e si noti inoltre come già sommando poche armoniche si ottenga un'onda che già assomiglia molto ad un'onda quadra. Senza addentrarci nella matematica, questa definizione ci è sufficiente per capire che l'onda quadra a 50Hz con cui viene pilotato il trasformatore di figura 1 può essere pensata come una moltitudine di onde sinusoidali di frequenza 50Hz, 100Hz, 150Hz e così via (più precisamente un'onda quadra è composta dalle sole armoniche di ordine dispari quindi 50Hz, 150Hz, 250Hz…). Di tutte queste onde, la fondamentale (di frequenza 50Hz) è quella che ha ampiezza maggiore ed è quella che determina il corretto funzionamento del trasformatore.

Le armoniche di ordine superiore hanno ampiezza trascurabile rispetto alla fondamentale ed il loro effetto è attenuato anche grazie all'induttanza dell'avvolgimento che funziona da filtro. Ecco il motivo per cui in uscita al trasformatore si ha un segnale sinusoidale e non un'onda quadra. Viene utilizzata un'onda quadra in quanto questo tipo di segnale è assai facile da generare usando un multivibratore astabile digitale. Generare un segnale sinusoidale a 50Hz sarebbe stato molto più complesso.

LA MODULAZIONE PWM

Grazie alla modulazione PWM si possono generare segnali analogici partendo dalle uscite digitali di un microcontrollore. Ecco come fare

Quella di generare o controllare segnali analogici utilizzando un microcontrollore è un'esigenza abbastanza frequente. In molti casi infatti un'informazione binaria tipo acceso/spento non è sufficiente, ma occorrere un segnale analogico vero e proprio al cui livello possa essere associata un'informazione. Quasi tutti i microcontrollori mettono a disposizione dei convertitori Analogico/Digitali, ma quasi nessuno incorpora la funzione opposta. Una soluzione un po' drastica (dal punto di vista dei costi) potrebbe essere quella di utilizzare un convertitore Digitale/Analogico esterno. Per fortuna nella maggior parte di applicazioni non occorre arrivare a tanto, ed è possibile utilizzare i dispositivi messi a disposizione dal micro per ottenere questa funzione. Per fare questo si ricorrere infatti alla modulazione PWM (*Pulse Width Modulation*, ossia modulazione a larghezza di impulsi), che consente appunto di generare dei segnali analogici partendo direttamente da segnali digitali. L'idea è la seguente: si utilizza un'onda quadra, di solito a frequenza costante, scegliendo opportunamente il tempo in cui essa si mantiene a livello alto in ciascun periodo. Questo tempo è quello in cui in ogni ciclo effettivamente forniamo energia al dispositivo collegato, nel tempo rimanente questa energia verrà dissipata (o utilizzata) dal dispositivo stesso. L'energia media su un periodo sarà quindi funzione del rapporto tra la durata del livello alto e quella del livello basso (questo rapporto è chiamato "*duty cycle*"). Se vogliamo ottenere un valore "medio" stabile occorrerà che il dispositivo collegato sia "sufficientemente lento", in modo da eseguire proprio l'operazione di "media" nel tempo. Nel caso di un microcontrollore (o altro dispositivo digitale) potremo ottenere un valore di tensione analogico compreso tra circa 0V e 5V (o comunque la tensione del

Figura 1 Generazione di un segnale PWM.

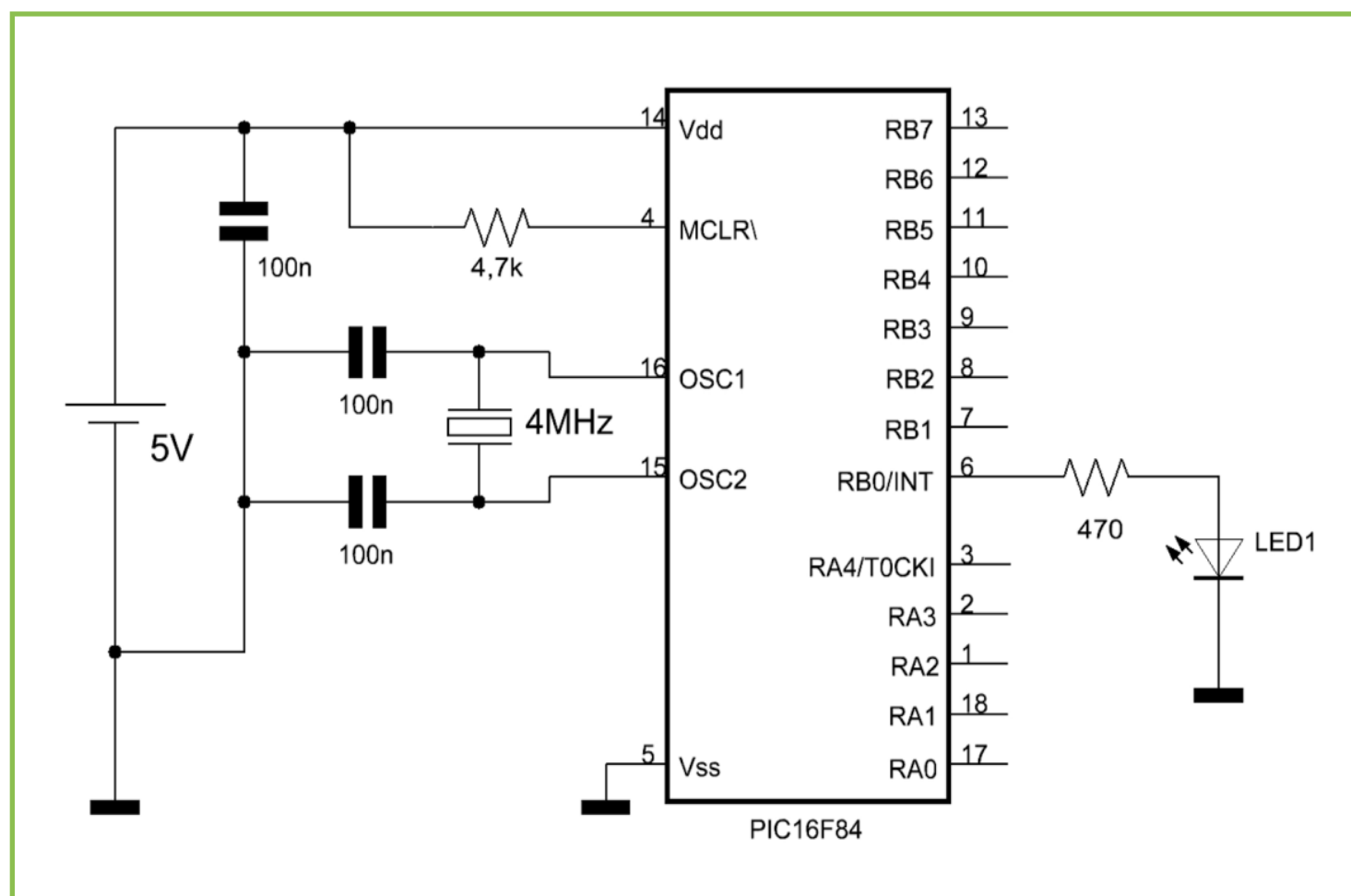



Figura 2 Schema per l'accensione del LED con onda PWM.

livello logico alto), semplicemente generando un'onda quadra e variando il duty cycle dallo 0% al 100%. Come già detto è necessario che il circuito collegato a valle abbia un comportamento "passa basso", in modo da rispondere con sufficiente lentezza. Se questa condizione non è verificata, basta aggiungere un filtro passa basso passivo come mostrato in seguito. Ma come si genera un segnale PWM? Il metodo classico è illustrato in Figura 1. È utile comprendere questo metodo perché sarà quello che utilizzeremo negli esempi seguenti (implementato via software o hardware). Si utilizza un contatore che viene costantemente incrementato, ed un valore di soglia: se il valore del contatore è inferiore alla soglia, l'uscita sarà a livello alto, se il contatore è maggiore o uguale, l'uscita sarà posta a livello basso. Per valori bassi della soglia si otterranno quindi duty cycle bassi, e viceversa. Nei prossimi paragrafi vedremo come implementare questo meccanismo in C, utilizzando diverse tecniche.

PWM CON CODICE SEQUENZIALE

Il codice riportato di seguito mostra com'è possibile implementare in C un semplice generatore di segnale PWM. In questo caso il segnale viene costruito letteralmente come spiegato prima: viene usata una variabile per implementare il contatore, una per la soglia ed una condizione che aggiorna il livello dell'uscita. Questo approccio ha il vantaggio di essere utilizzabile su qualsiasi PIC, anche i più semplici, in quanto non si ricorre ad alcuna periferica hardware specifica, ed è possibile avere l'uscita su qualsiasi piedino di I/O. Il segnale PWM generato viene utilizzato nello schema di Figura 2 per fare lampeggiare un LED con una variazione graduale della luminosità. Ci si potrebbe chiedere come mai non sia stato usato un filtro passa basso. In effetti il LED è un dispositivo abbastanza veloce, e non esegue affatto la media del segnale, ma ne segue fedelmente il valore. Tuttavia il risultato che si ottiene è buono, perché la media è eseguita dal nostro occhio, che ha un tempo di risposta di circa un ventesimo di secondo! Il codice è riportato nel listato 1. La variabile cont è usata come contatore, ed è quindi incrementata ad ogni iterazione. Dal momento che è ampia 8 bit, raggiunto il limite di 255 tornerà automaticamente a zero. La variabile *level* è usata come soglia. Se avessimo assegnato un valore fisso, avremo potuto ottenere un valore stabile di luminosità del LED. In questo caso invece il valore della soglia è variato progressivamente, in modo da avere un duty cycle che inizia dallo 0% *(level=0)* al 100% *(level=255)*. La soglia è incrementata ogni 200 iterazioni, in modo da rendere visibile la variazione, per fare questo è stata utilizzata la variabile *i*, che serve proprio per contare quante volte viene eseguito il loop. Un paio di considerazioni importanti: quanti livelli di luminosità possiamo ottenere? Ben 256, cioè tanti quanti sono i livelli del contatore PWM. Qual è invece la frequenza della nostra onda PWM? Qui la risposta è meno ovvia: per completare un periodo è necessario che il contatore esaurisca il conteggio, quindi serviranno ben 256 iterazioni del loop per completare un solo periodo!

LISTATO 1

```
void main() {
unsigned char cont, level, i;
TRISB = 0;
PORTB = 0;
cont=0;
level=0;
i=0;
// Loop infinito
while(1) {
if (cont>level) {
PORTB=0;
} else {
PORTB=1;
}
cont++;
i++;
if (i==200) {
level++;
i=0;
}
}
}
```

Supponendo di utilizzare un PIC funzionante a 4MHz, possiamo misurare quanto tempo impieghiamo per eseguire un'iterazione del loop utilizzando il Debugger del MikroC, facendolo funzionare in modalità step, e osservando il tempo di esecuzione riportato in basso. Il risultato è 16us, quindi 16us*256=4096us, cioè ben 4ms (244Hz)! Giusto in tempo per non percepire un fastidioso sfarfallio…Questo semplice calcolo mette in luce un problema tipico del PWM: se vogliamo ottenere una risoluzione alta, dobbiamo accontentarci di frequenze basse, a meno di non "contare"

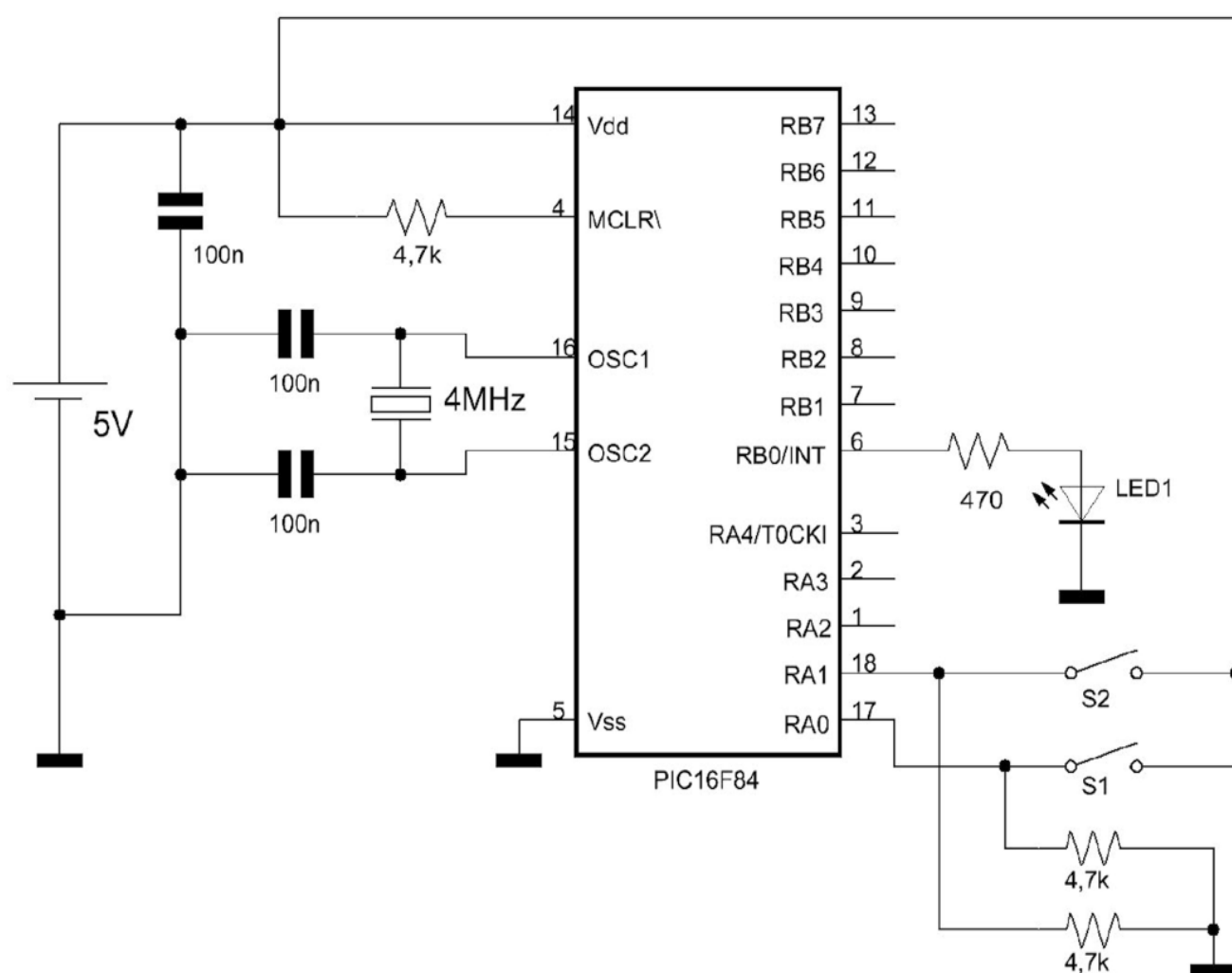



Figura 3 Schema per controllo luminosità LED.

a velocità molto elevate (cosa non sempre possibile o conveniente).

PWM CON INTERRUZIONI

Uno dei problemi del codice presentato prima è che il periodo dell'onda PWM dipende dalla quantità delle istruzioni che sono presenti nel loop principale. Ovviamente in programmi completi il loop sarà affollatissimo di istruzioni e routine e quindi la frequenza del PWM risulterà bassissima e probabilmente non costante. Per risolvere questo inconveniente, come abbiamo fatto in altre occasioni, possiamo utilizzare le interruzioni. Programmiamo il timer TMR0 del PIC in modo da generare un'interruzione ad intervalli regolari, ed utilizziamo la routine di servizio per incrementare il contatore PWM ed aggiornare l'uscita. In questo caso la frequenza dell'onda generata sarà facile da calcolare essendo data dalla frequenza delle interruzioni diviso il numero di passi del contatore. Questa frequenza risulterà inoltre indipendente dal contenuto del loop principale e dal resto del programma. Il codice riportato nel listato 2 implementa il metodo appena descritto: il segnale PWM viene utilizzato per regolare la luminosità di un LED, il cui livello può essere regolato manualmente agendo sui due pulsanti SW1 ed SW2, che permettono di aumentarlo o diminuirlo (lo schema è riportato in Figura 3). Si può notare come la gestione dei pulsanti sia fatta nel loop principale, dove per evitare problemi di rimbalzo è stato inserito addirittura un ritardo di 100ms! Tutto questo non interferisce con la generazione del segnale PWM, che invece segue tempistiche indipendenti. Un'altra differenza rispetto al codice presentato prima risiede nel fatto che il segnale PWM generato ha una risoluzione di 16 livelli e non 256. Questo è stato ottenuto incrementando il contatore di 16 invece che di 1 (l'overflow si ha in 256/16=16 passi). Questo accorgimento, unito all'utilizzo delle interruzioni permette di ottenere una frequenza PWM molto più alta. I pin RA0 e RA1 della porta A sono impostati come ingressi ed utilizzati per rilevare la pressione dei pulsanti. La porta B è utilizzata per la generazione dell'onda PWM (sul piedino RB0). Sono abilitate le interruzioni relative al TMR0, ed impostati il suoi parametri (clock interno, niente prescaler, etc.). Nel loop viene soltanto rilevata la pressione di uno dei due pulsanti ed aggiornato il livello, nella routine d'interruzione invece si trova lo stesso codice che prima era nel loop principale. Come già detto il contatore è incrementato di 16 unità invece che di 1, per ottenere una risoluzione di 16 livelli (4 bit). Si sarebbe anche potuto resettare il contare stesso una volta raggiunto il valore 16, in questo caso anche il livello sarebbe stato espresso come un numero compreso tra 0 e 15.

LISTATO 2

```
unsigned char cont, level;
void main() {
// Inizializz. porte
TRISA = 0x03;
PORTA = 0x00;
TRISB = 0x00;
// Inizializz. Registri
OPTION_REG = 0x80;
TMR0 = 0;
INTCON = 0xA0;
cont=0;
level=0;
// Loop infinito
while(1) {
if (PORTA&0x01)
level+=16;
if (PORTA&0x02)
level-=16;
Delay_ms(100);
}
}
void interrupt()
{
if (cont>level) {
PORTB=0;
} else {
PORTB=1;
}
cont=cont+16;
TMR0 = 0; // reset timer
INTCON = 0x20; // cancella flag
IRQ
}
```

GENERAZIONE PWM HARDWARE

Nonostante i vantaggi di utilizzare le interruzioni, la generazione del segnale PWM via software costituisce sempre un compi-

to piuttosto gravoso per il PIC, soprattutto quando deve svolgere anche diversi altri compiti o la frequenza scelta è piuttosto alta. Una soluzione molto efficiente è costituita dall'impiego di periferiche hardware apposite presenti su molti PIC. In questo caso è sufficiente programmare il modulo appo sito ed il segnale verrà generato in maniera del tutto indipendente dal software. Questo approccio come vedremo è reso più semplice dal fatto che il MikroC mette a disposizione alcune funzioni di libreria per utilizzare queste periferiche. Nel listato 3 viene utilizzato il modulo CCP (Capture / Cpmpare / PWM) di un PIC16F876 per controllare l'accensione e lo spegnimento "soft" di una lampadina a 12V (Figura 4). Va notato che quando si utilizza il generatore hardware l'uscita del segnale può essere prelevata soltanto dal piedino RC2 (non è selezionabile arbitrariamente). La funzione Pwm_Init imposta la frequenza dell'onda PWM, la funzione Pwm_Start avvia la generazione del segnale (esiste anche Pwm_Stop per fermarlo), mentre Pwm_Change_Duty serve per cambiare il valore di duty cycle generato. Come si vede, utilizzando queste funzioni il codice risulta particolarmente semplice. Ad ogni pressione del pulsante viene attivato l'aggiornamento del livello *(run=1)* ed invertita la direzione d'incremento (il modo da ottenere una rampa in salita ed in discesa in maniera alternata). Una volta raggiunto uno dei due estremi (0 o 255) l'aggiornamento viene arrestato *(run=0)*. Il ritardo di 20ms serve per evitare fenomeni di rimbalzo e per rendere più visibile la variazione di luminosità.

GENERAZIONE DI SEGNALI ANALOGICI

La tecnica PWM può essere utilizzata per generare segnali analogici. Per fare questo è sufficiente aggiornare il duty cycle dell'onda PWM a intervalli regolari, con i campioni del segnale analogico da riprodurre. La frequenza di riproduzione dei campioni può essere al massimo pari a quella dell'onda PWM, e quindi la massima frequenza analogica riproducibile risulta metà di questa frequenza. In realtà per evitare distorsioni è bene fare in modo che ogni campione della forma d'onda analogica sia mantenuto per diversi cicli dell'onda PWM. Il codice nel listato 4 mostra la generazione di una sinusoide utilizzando la tecnica descritta, ed il circuito di Figura 5. I 16

Figura 4 Schema per accensione e spegnimento soft di una lampada.

LISTATO 3

```
void main() {
unsigned char level;
char delta, run;
// Iniz. porte
TRISB = 0x01;
PORTB = 0;
PORTC = 0;
TRISC = 0;
// Init. modulo PWM
Pwm_Init(1000);  // Freq.  PWM
1000Hz
Pwm_Start(); // Start PWM
delta=1;
run=0;
level=255;
// Loop infinito
while(1) {
Pwm_Change_Duty(level);
if (PORTB) {
delta=-delta;
run=1;
while(PORTB){};
}
if (run) level=level+delta;
if ((level==0)||(level==255))
run=0;
Delay_ms(20);
}
}
```

LISTATO 4

```
unsigned char j;
const unsigned char
onda[16]={128,176,218,246,
255,246,218,176,
128, 79,
37, 10,
0, 10, 37,
79};
void main() {
// Inizializz. porte e regs
PORTC = 0;
TRISC = 0;
OPTION_REG = 0x80;
TMR0 = 192;
INTCON = 0xA0;
// Init. modulo PWM
Pwm_Init(31248); // Freq. PWM =
31KHz
Pwm_Start(); // Start PWM
// Loop infinito
while(1) {}
}
void interrupt()
{
Pwm_Change_Duty(onda[j&0x0F]);
j++;
TMR0 = 192; // reset timer
INTCON = 0x20;  // reset flag
IRQ
}
```

Figura 5 Schema per la generazione di un segnale analogico.

campioni della sinusoide sono memorizzati in un array, che ad ogni interruzione viene letto per assegnare al duty cycle il valore del campione corrente (puntato dall'indice j). La frequenza dell'onda generata sarà data da quella di overflow del TMR0 (Fclk/4 diviso 256-192=64), diviso 16 campioni per periodo. Partendo da un clock di 8MHz si ottiene una sinusoide della frequenza di circa 300Hz. Per aumentare la frequenza della sinusoide occorre caricare in TMR0 un valore maggiore, in modo da aumentare la frequenza con cui vengono generate le interruzioni, ma occorre mantenersi sotto la frequenza dell'onda PWM, che è stata scelta pari a 31248Hz. Per ottenere un segnale analogico pulito (poco distorto) si dovrebbe fare in modo da mantenere la frequenza del segnale generato abbastanza più bassa di quella di campionamento, o in caso contrario utilizzare un filtro passa basso abbastanza ripido la cui frequenza di taglio può essere più vicina a quella di campionamento.
Nel circuito di Figura 5 è stato utilizzato un semplice circuito RC come filtro passa-basso, la cui frequenza di taglio è circa 1500Hz. Va ricordato che in questo caso il carico deve avere un'impedenza sufficientemente alta (almeno qualche Kohm), in modo da non variare la costante di tempo del circuito.

GENERAZIONE DI SUONI ED EFFETTI

Apparentemente l'argomento potrebbe sembrare poco correlato con la generazione di segnali PWM, invece proprio i componenti del PIC che generano i segnali PWM possono essere utilizzati anche per generare suoni, note musicali ed effetti sonori. Agendo sui rispettivi registri infatti è possibile modulare sia l'ampiezza (attraverso il duty cycle) che la frequenza del segnale generato. Una volta inizializzato ed avviato il modulo PWM è possibile variare il duty cycle agendo sul registro CCPR1L, e la frequenza, agendo sul regi stro PR2. Il primo determina il valore del contatore TMR2 superato il quale l'uscita si porta a livello basso, mentre il secondo determina il valore per cui il timer viene resettato (più piccolo sarà questo valore, più frequente sarà l'overflow). Se vogliamo usare il modulo PWM per generare suoni dobbiamo soltanto tenere presente due cose: 1) il prescaler di TMR2 (bit 0 e 1 del registro T2CON) deve essere impostato in modo che la frequenza ottenuta risulti udibile; 2) Il valore di duty cycle deve essere inferiore al limite di conteggio, altrimenti non ci saranno variazioni nel segnale d'uscita. Maggiori dettagli possono essere trovati nel datasheet del PIC. Nell'esempio riportato nel listato 5 è stato utilizzato un PIC16F876 con clock ad 8MHz (stesso schema di Figura 5), il prescaler è stato impostato a 1:16, ottenendo frequenze che coprono un range che va da circa 100Hz a circa 1KHz. Il programma assegna dei valori crescenti e poi decrescenti al registro PR2, ad intervalli di 100ms, ottenendo un effetto sonoro tipo scala ascendente e discendente (o "sirena", se riprodotto più velocemente). E' possibile anche calcolare i valori per PR2 in modo da ottenere le note musicali, che suonate in sequenza possono produrre una melodia vera e propria. Invece assegnando a PR2 una sequenza veloce di valori "arbitrari" si possono ottenere degli interessanti effetti sonori (molto simili a quelli dei vecchi videogame!). Com'è facile immaginare anche in questo caso l'aggiornamento dei valori può essere fatto all'interno di una routine d'interruzione, in modo da lasciare spazio ad altre routine all'interno del loop principale.

LISTATO 4

```
void main() {
unsigned char j;
// Init. porte
PORTC = 0;
TRISC = 0;
Pwm_Init(1000);
Pwm_Change_Duty(128);
Pwm_Start();
T2CON=0x07; // Init. prescaler
// Loop infinito
while(1) {
for(j=70; j<255; j+=5) {
PR2=j;
Delay_ms(100);
}
for(j=255; j>70; j-=5) {
PR2=j;
Delay_ms(100);
}
}
}
```

di Angelo De Bartolo

RICONDIZIONARE UN CONTATTO D'ALLARME

Alcuni produttori di sensori per antintrusione preferiscono bilanciare internamente i loro dispositivi ma ciò rende spesso incompatibile i sensori con le altre centrali. Questo circuito ricondiziona un contatto di allarme internamente bilanciato in un contatto "pulito"

Un installatore di sistemi di antintrusione qualche giorno fa mi diede un sensore EV100-CS della General Electric divisione Security provvisto di telecamera interna a colori affidandomi il compito di renderlo "compatibile" con le centrali di un altro produttore in quanto non utilizzabile allo stato attuale.
Normalmente il costo di un sensore PIR (presenza ad infrarossi) è molto contenuto e quindi non varrebbe la pena stare qui a discutere di un sistema di condizionamento ma questo sensore specifico ha un valore maggiore dato che incorpora all'interno anche una microtelecamera; inoltre, visto il gran numero di esemplari posseduti dall'installatore, può essere economicamente vantaggioso studiare un semplice sistema di compatibilità.

IL BILANCIAMENTO NEI SISTEMI DI ANTINTRUSIONE

I moderni sistemi di antintrusione adoperano un bilanciamento della linea (spesso chiamata "*zona*") al posto del semplice "*contatto normalmente chiuso N.C.*" in quanto garantisce un maggior livello di protezione da sabotaggi.
Nei sistemi di antintrusione non bilanciati basterebbe cortocircuitare la linea con un ponticello per eludere il controllo di presenza.
Nelle linee bilanciate, invece, un valore di resistenza viene preso come riferimento per la condizione di riposo ed un cortocircuito o uno sbilanciamento della resistenza verrebbe visto come una condizione anomala e pertanto verrebbe segnalata dalla centrale. In generale una linea di antintrusione può essere:

- *non bilanciata*, o N.C.;
- *a singolo bilanciamento*, il contatto di allarme è bilanciato a fine linea (E.O.L, End Of Line) tramite una resistenza di opportuno valore, mentre il contatto del tamper viene posto in serie con i tamper degli altri sensori;
- *a doppio bilanciamento*, il contatto di allarme e il contatto di antimanomissione (tamper) vengono collegati in serie tramite una resistenza, infine una resistenza di fine linea viene utilizzata per il bilanciamento: ciò comporta la concentrazione di entrambi i segnali su un'unica coppia di conduttori (giallo e verde in Figura 1);
- *triplo bilanciamento*, il contatto di allarme, il contatto di antimanomissione ed il contatto di antimascheramento vengono collegati fra loro tramite opportune resistenze.

In figura 1 sono rappresentate le prime tre condizioni sopra elencate.

Condizioni di allarme

Il metodo più utilizzato dagli installatori è il collegamento a doppio bilanciamento sia

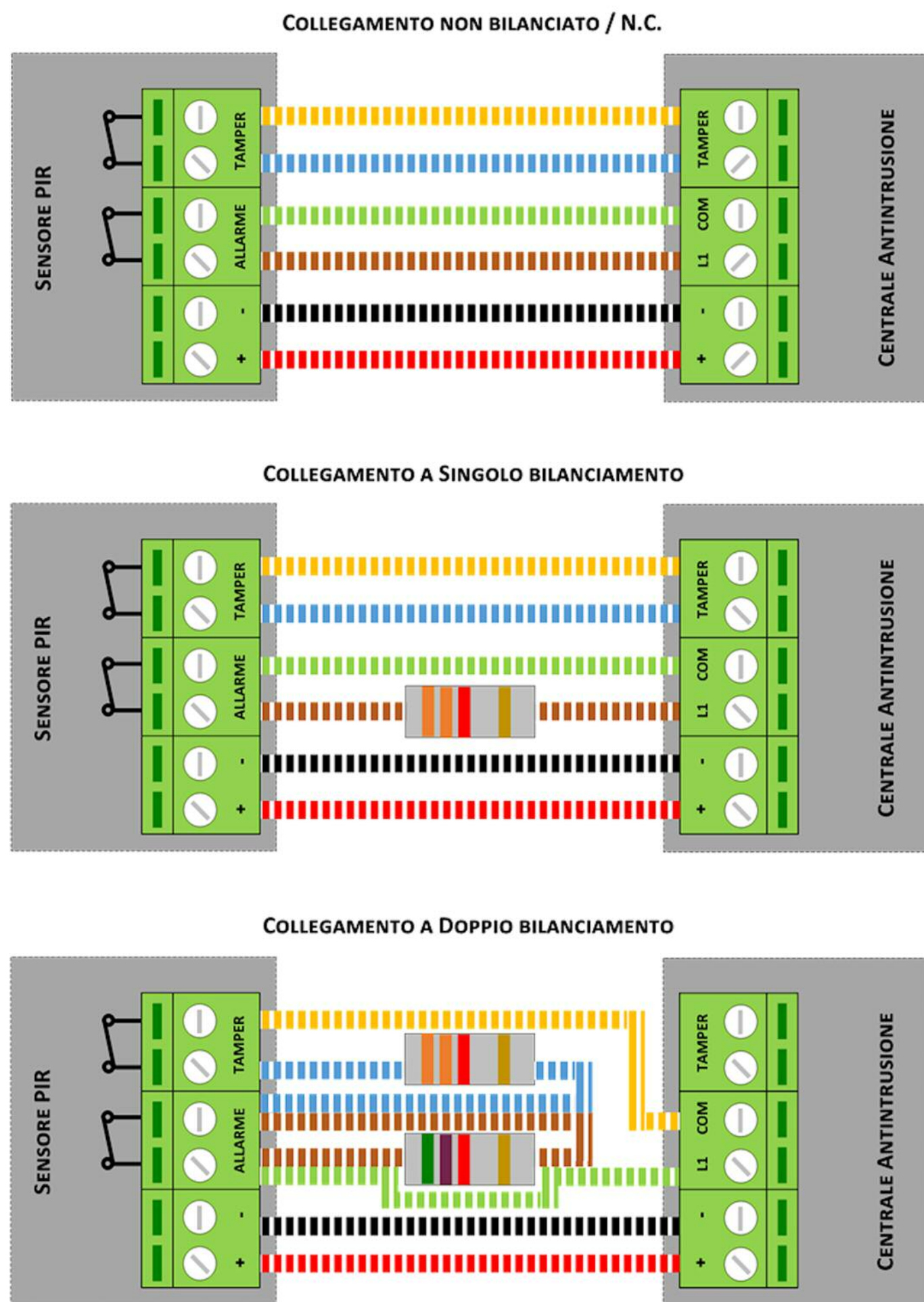



Figura 1: Tipologie di collegamento dei sensori ad una centrale antintrusione

perché assicura un maggiore grado di protezione dai sabotaggi rispetto ad un collegamento non bilanciato e sia perché semplifica (rispetto al singolo bilanciamento) i collegamenti in centrale e riduce il numero di conduttori necessari: il triplo bilanciamento è effettuabile solo con sensori provvisti di un terzo contatto di antimascheramento.

In un sistema collegato a doppio bilanciamento la condizione di riposo equivale ad una resistenza equivalente coincidente con la resistenza di bilanciamento in serie tra il contatto denominato Tamper (antimanomissione) e il contatto d'allarme; l'apertura del cover del sensore comporterà l'aprirsi del microcontatto Tamper il che viene "visto" in centrale come un'interruzione della linea; analogamente un taglio del cavo di collegamento del sensore viene visto come un'interruzione della linea ed anche in questo caso verrà segnalata una manomissione.

Una condizione di allarme, a seguito del rivelamento di una presenza, comporta l'aprirsi del microcontatto denominato Allarme, in questa condizione la centrale vedrà su quella linea una resistenza totale che è somma delle due resistenze di bilanciamento.

E' importante precisare che il bilanciamento va fatto ai morsetti del sensore e non in centrale altrimenti viene meno la protezione contro una manomissione con corto circuito.

Il bilanciamento nel sensore EV100-CS

E' doveroso precisare che il nuovo modello del sensore EV100-CS (distribuito da UTC Fire & Security che ha acquisito G.E. Security) non ha il bilanciamento interno ma bensì lavora con relè a contatti liberi mentre il modello su cui ho lavorato è una vecchia serie, pur avendo lo stesso codice prodotto.

Leggendo il datasheet del produttore (figura 3) ci rendiamo conto che il sensore possiede già un doppio bilanciamento interno costituito da due resistori da 4,7kΩ pertanto le possibili condizioni sono:

- *condizione di riposo*, resistenza equivalente da 4,7kΩ
- *condizione di allarme presenza*, resistenza equivalente 4,7kΩ + 4,7kΩ = 9,4kΩ
- *condizione di allarme tamper*, resistenza equivalente ad un circuito aperto, che si realizza fisicamente togliendo l'alimentazione al sensore.

IL CIRCUITO REALIZZATO

L'idea alla base del circuito è un sistema capace di discriminare il valore di resistenza di un doppio partitore di tensione usato come ingresso dei morsetti di ingresso dell'operazionale che lavora a loop aperto in tipica configurazione da comparatore.

Per rendere agevole la lettura del valore di resistenza si è ragionato in termini di tensioni, ovvero il morsetto del sensore è stato collegato all'alimentazione del sensore stesso (solitamente 12V÷13V) tramite una resistenza serie di opportuno valore.

Verifichiamo le varie condizioni in cui può trovarsi il sensore ed il comportamento del circuito.

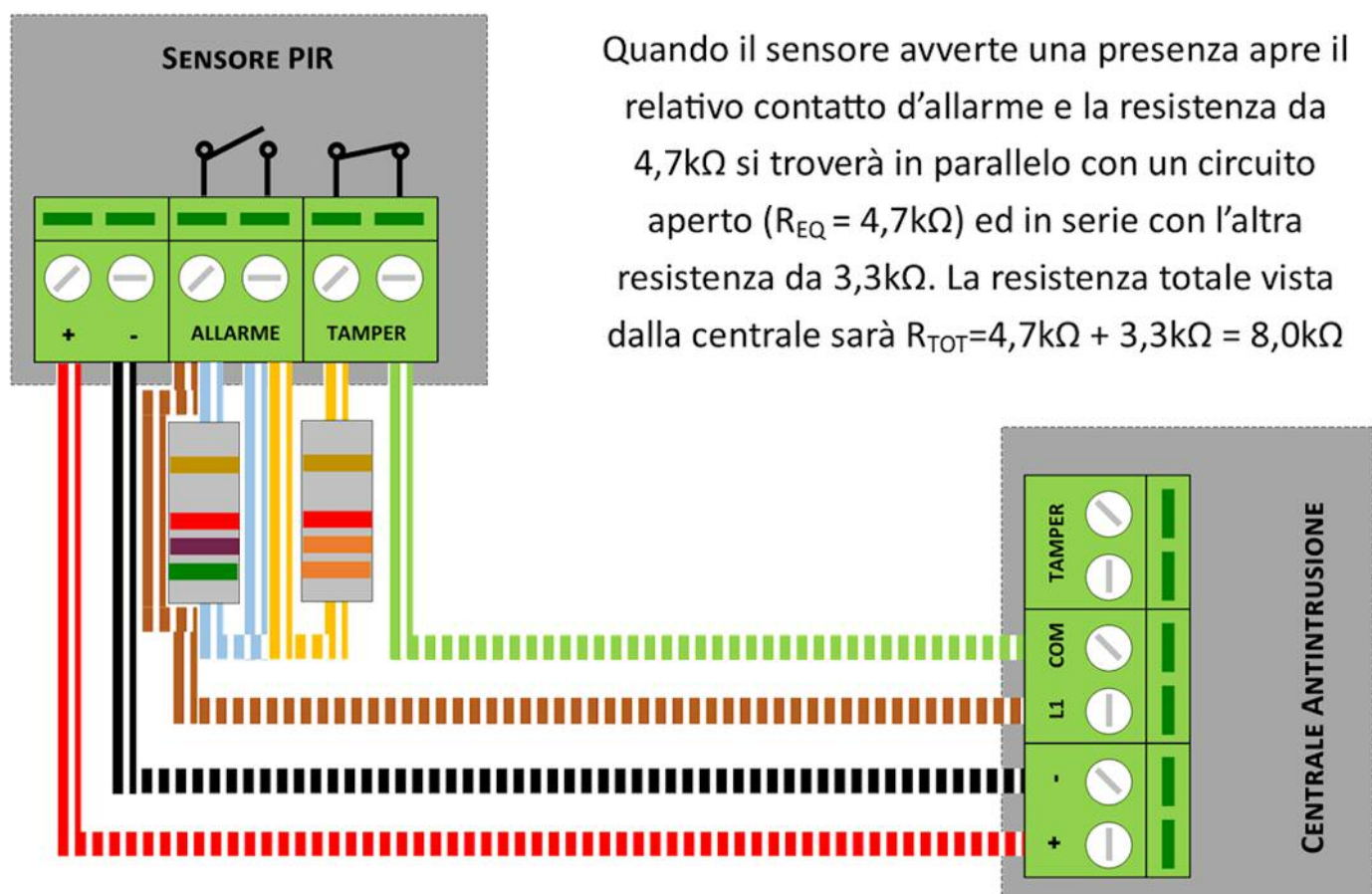
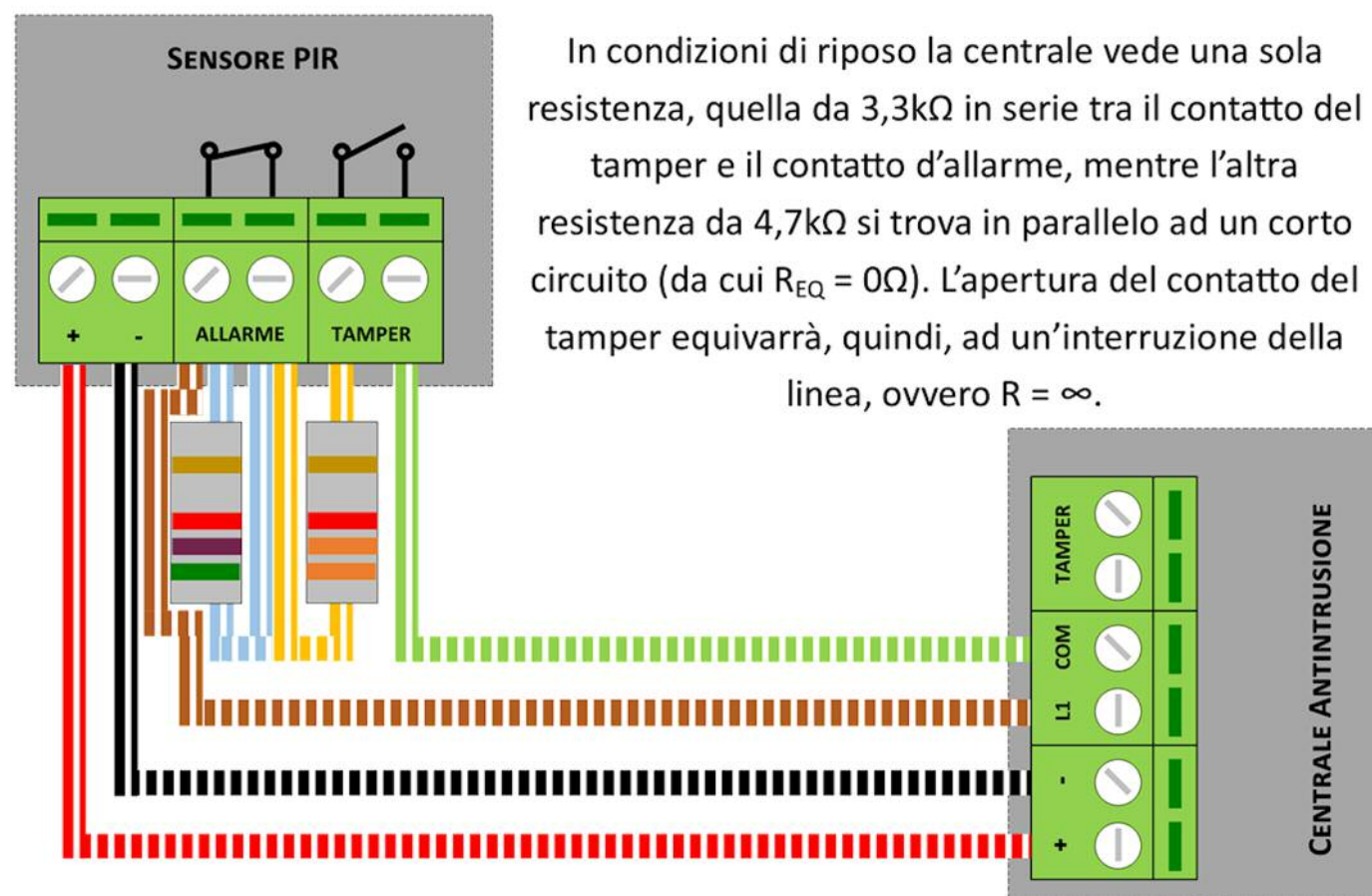



Figura 2: Resistenza equivalente nelle condizioni di allarme

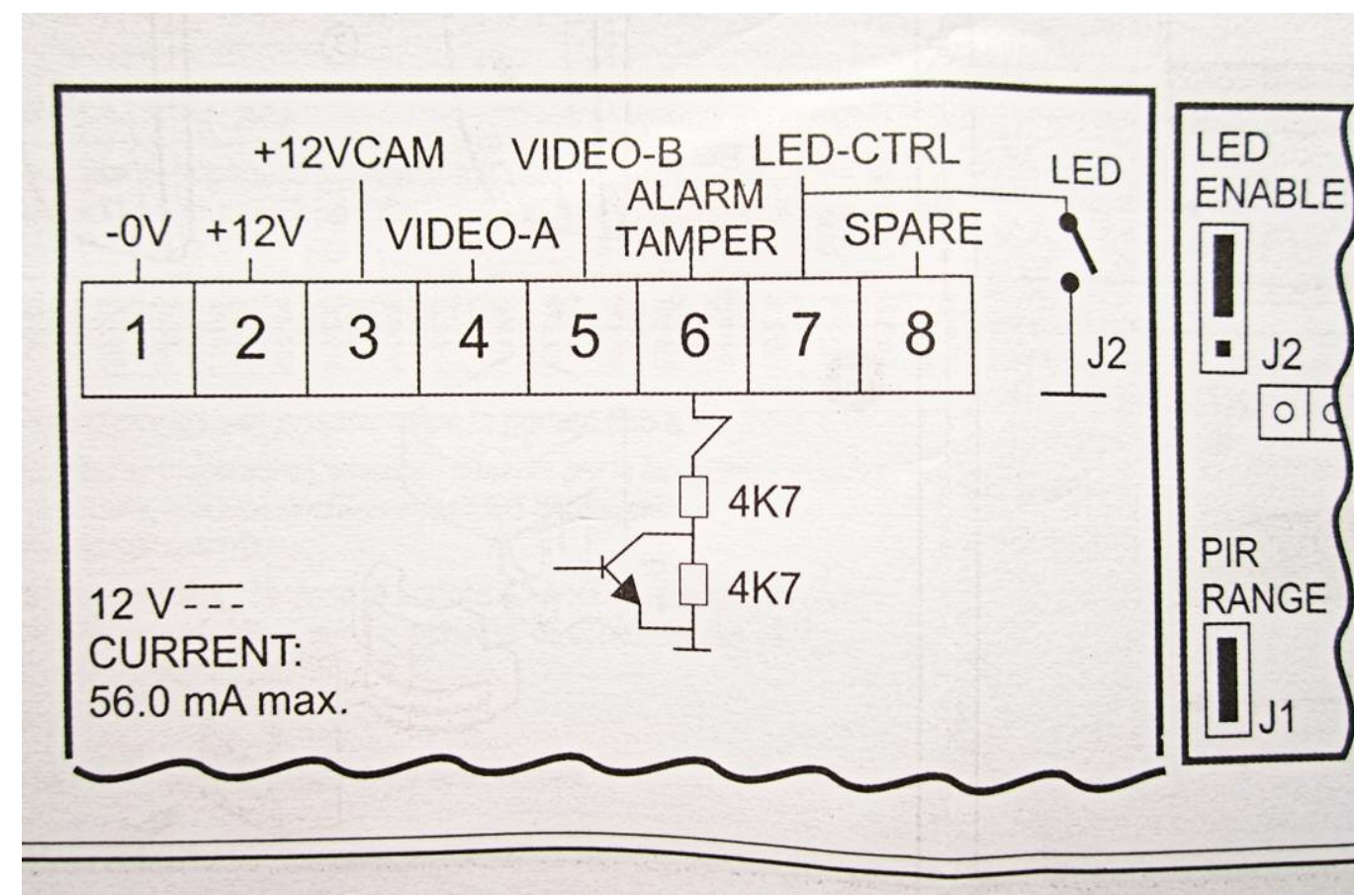



Figura 3: Estratto del datasheet del sensore EV100-CS

Condizione di riposo

In condizioni di riposo la tensione presente sul morsetto non invertente dell'operazione è maggiore della tensione presente sul morsetto invertente in quanto il sensore PIR (schematizzato con R4, R5 ed un pulsante N.C) avrà una resistenza equivalente di circa 4,7kΩ.
Ciò comporta che l'uscita dell'operazionale si attesta ad un livello vicino al valore di alimentazione positiva (in base all'operazionale scelto sarà compreso tra 9V e 12V) che è sufficiente a pilotare un MOSFET a canale N tramite una resistenza che entrando in conduzione chiuderà il relè verso massa permettendo alla bobina di eccitarsi e quindi commutare. E' stato scelto di tenere eccitata la bobina in posizione di riposo in modo tale che aprendo il cover del sensore viene a mancare l'alimentazione ed il contatto si aprirà segnalando un allarme in centrale.

Condizione di allarme

In condizioni di allarme il sensore PIR avrà una resistenza equivalente:

$$4{,}7k\Omega + 4.7k\Omega = 9.4k\Omega$$

Ciò comporterà che la tensione presente ai capi del morsetto invertente sarà maggiore della tensione presente ai capi del morsetto non invertente e quindi l'uscita dell'operazionale sarà prossima al valore di alimentazione negativo (che è 0V) e pertanto il MOSFET non entrerà in conduzione e la bobina sarà diseccitata.
Tutti i componenti del circuito non sono cri-

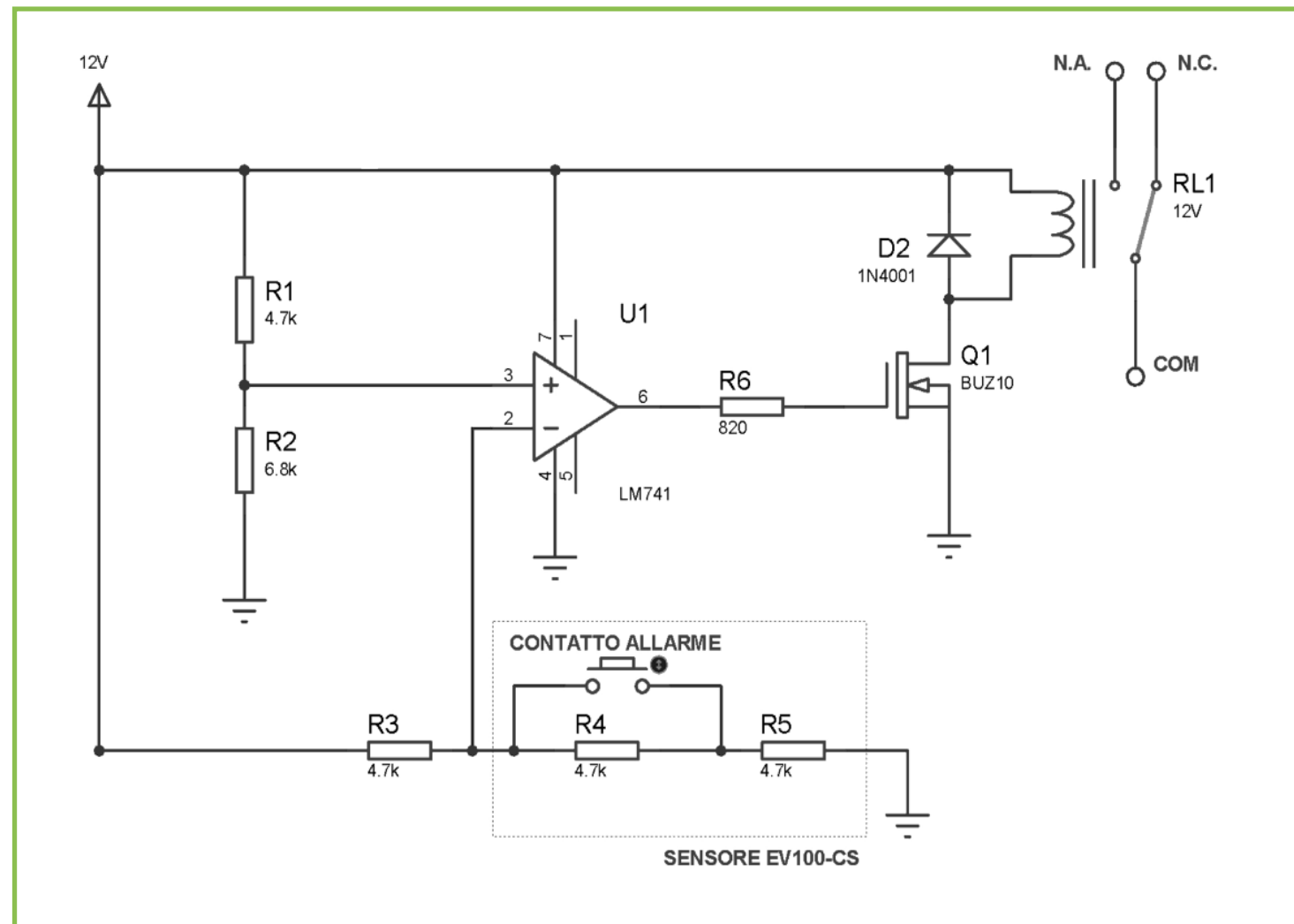



Figura 4: Circuito ricondizionatore di contatto

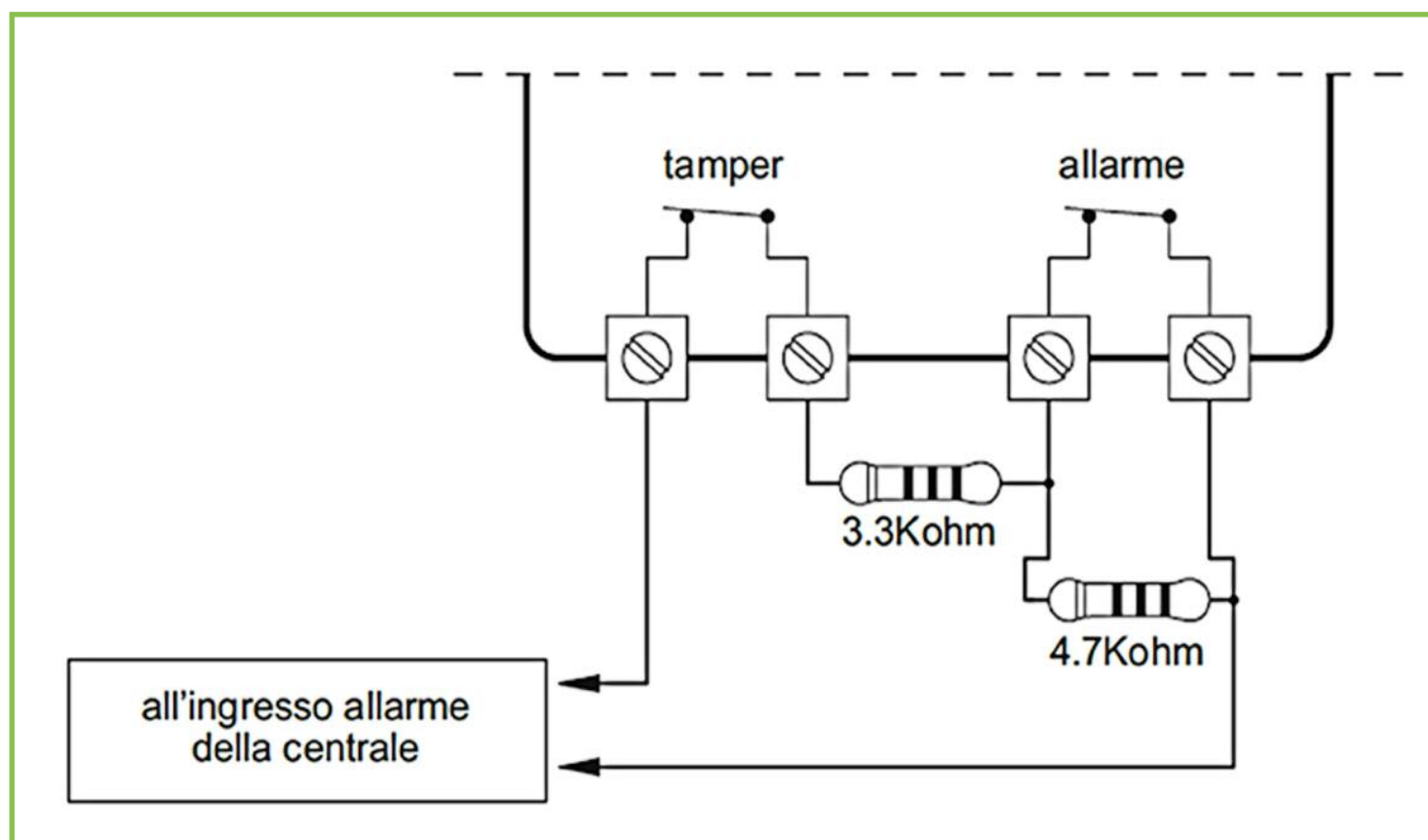



Figura 5: Estratto del manuale tecnico della centrale Axel Web Atlantis 32

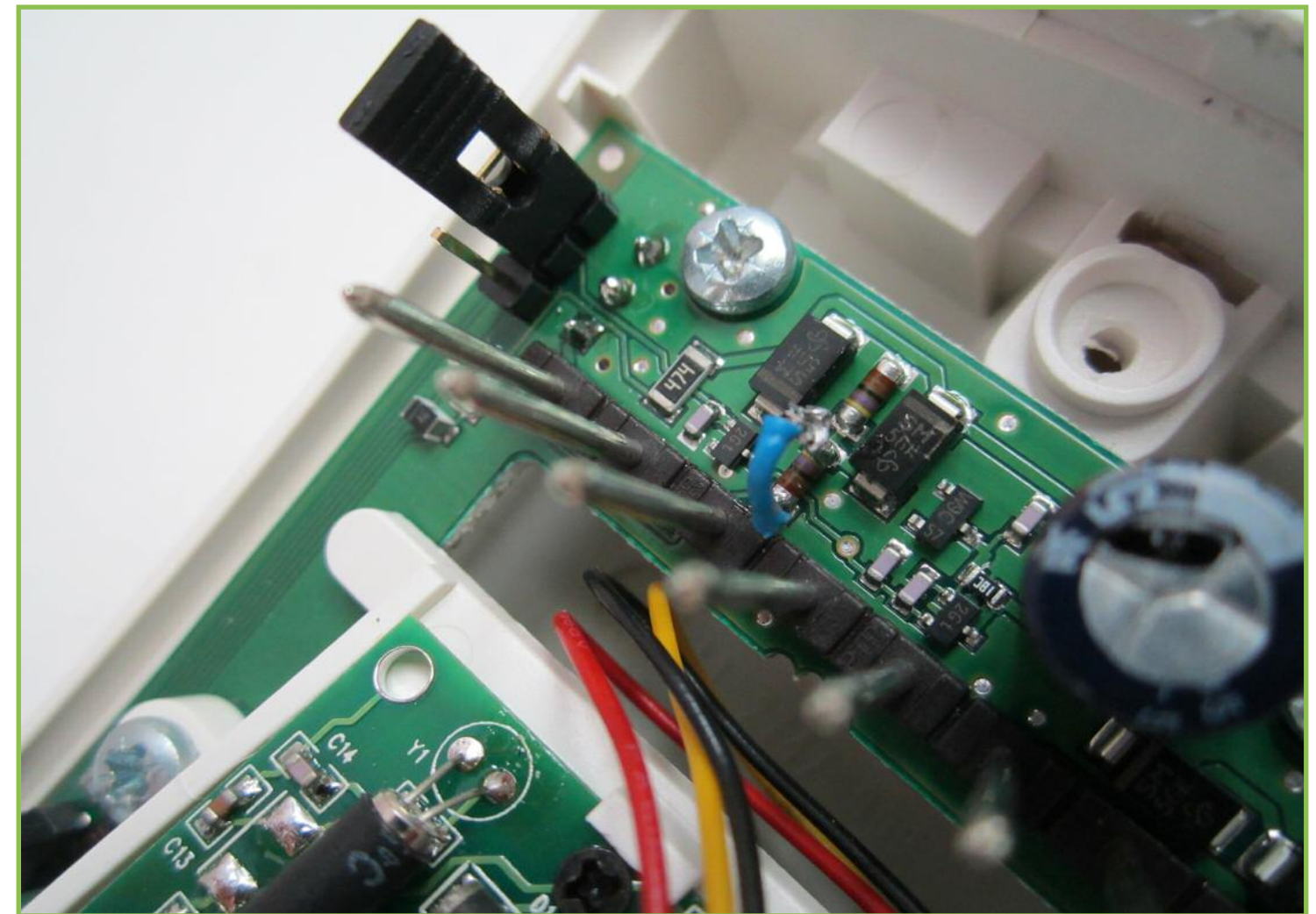



Figura 6: Modifica del PCB del sensore EV100-CS

tici nel senso che sono facilmente sostituibili con altri componenti più economici o semplicemente che si hanno disponibili: l'unico vincolo riguarda il doppio partitore di tensione (resistenze R1, R2, R3) i cui valori sono vincolanti. Il diodo in parallelo alla bobina del relè serve a limitare gli effetti delle extratensioni ("picco di Lenz").

La componentistica è stata volontariamente ridotta al minimo indispensabile per motivi di spazio.

L'ALTERNATIVA AL CIRCUITO: MODIFICA DEL PCB

L'alternativa è modificare il PCB del sensore: questa è la soluzione adottata nella pratica per motivi di spazio. In pratica sul PCB del PIR, vicino al pin 6 (uscita allarme e tamper) si notano due piccole resistenze in serie: la prima – più vicina al pin – sempre presente in quanto è la resistenza di bilanciamento del tamper, ed una seconda resistenza che "interviene" solo in caso di allarme.

Nel nostro caso specifico la centrale su cui dovranno essere montati i sensori accetta valori diversi di bilanciamento, più precisamente i valori accettati sono:

- resistenza da 3,3kΩ in condizioni di riposo
- resistenza somma di 3,3kΩ (riposo) + 4,7kΩ (allarme) in caso di allarme

Pertanto la seconda resistenza del valore di 4,7kΩ che interviene in condizioni di al-

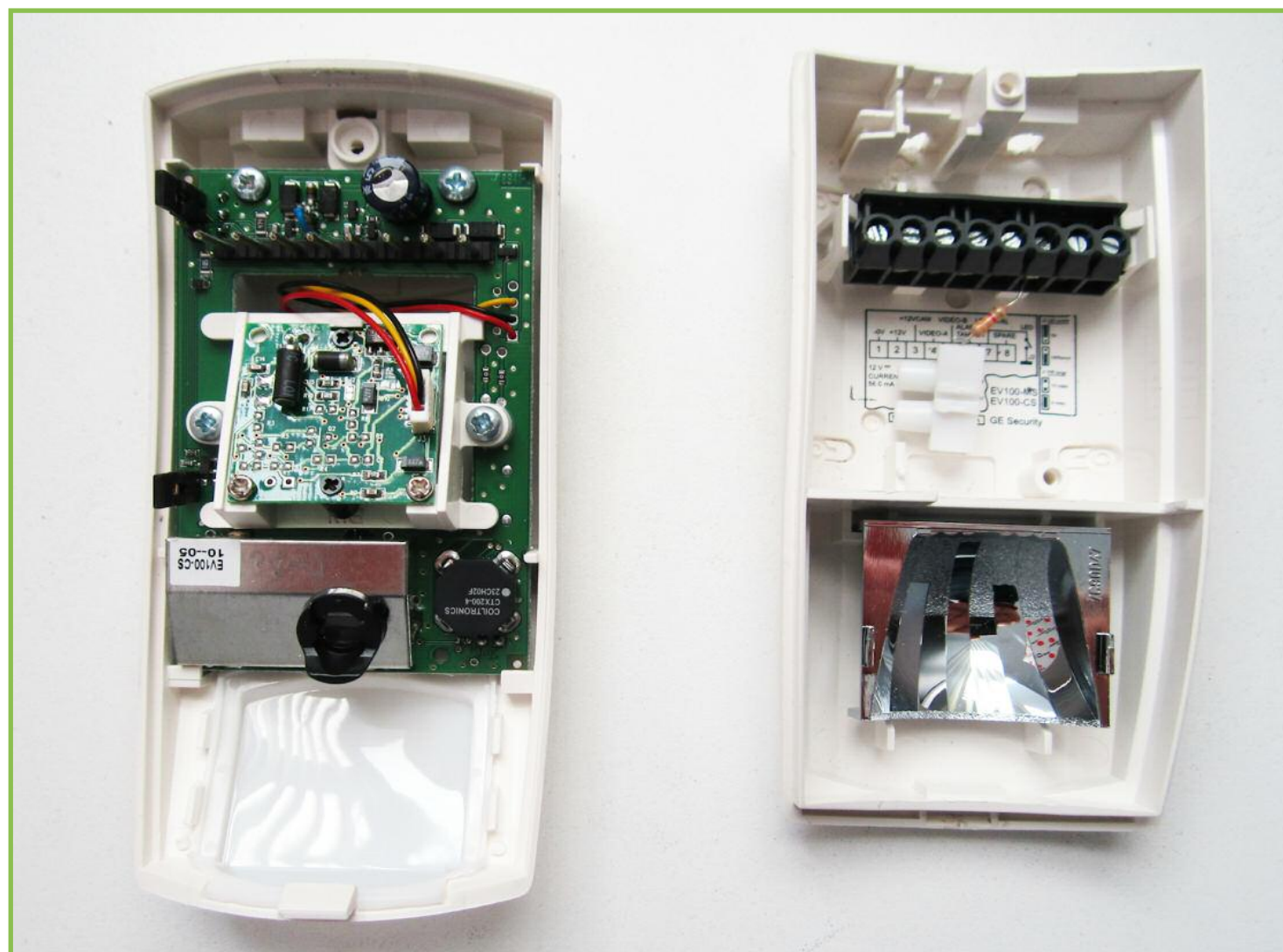



Figura 7: Vista del sensore e del cover

larme coincide con il valore della seconda resistenza nel nostro sensore e quindi non necessita di essere modificata.
La resistenza in condizioni di riposo, invece, non è compatibile e procederemo anzitutto a cortocircuitarla con un pezzetto di cavo (il cavetto blu in foto): così facendo in condizioni di riposo la resistenza equivalente è circa 0Ω.
Successivamente inseriremo una resistenza serie da 3,3kΩ (visibile sulla destra in figura 7) dopo il pin6 del morsetto del sensore in modo tale che in condizioni di riposo la resistenza vista dalla centrale sia proprio di 3,3kΩ.
L'effetto finale è visibile in figura 6 e 7.
Così facendo il tutto è stato risolto con una sola resistenza da 3,3kΩ ed un ponticello effettuato con un pezzetto di cavo di misura 0,22 AWG (corrispondente a 0,644 mm^2) recuperato da uno spezzone di cavo di rete UTP.

Elenco componenti	
R1	4,7KΩ 1/4 W
R2	4,8KΩ 1/4 W
R3	4,7KΩ 1/4 W
R6	820Ω 1/4 W
Q1	BUZ10
D1	1N4001
RL1	RELE' CON BOBINA A 12VDC
U1	LM741

PILOTARE UN MONITOR VGA

Ecco un interessante circuito basato su una FPGA, in grado di pilotare un monitor standard VGA

Nonostante l'avvento delle più moderne interfacce DVI e HDMI, lo standard VGA/VESA è ancora supportato da molti dispositivi di visualizzazione (monitor CRT, LCD, videoproiettori,etc.). Di solito questa si presenta come un connettore sub-D ad alta densità a 15 pin (Figura 1), in cui solo 5 sono utilizzati per comunicare il segnale video vero e proprio. In particolare i pin 1, 2 e 3 sono associati alle componenti di colore (rosso, verde e blu, in sigla R, G e B), mentre i pin 13 e 14 ai segnali di sincronismo orizzontale e verticale. I dati relativi all'immagine sono inviati con una scansione orizzontale (da sinistra a destra) e verticale (dall'alto al basso) come avviene nei sistemi televisivi. Dopo la fine di ogni linea segue un impulso di sincronismo orizzontale, mentre dopo la fine di ogni quadro segue un impulso di sincronismo verticale (i dettagli sulle temporizzazioni verranno forniti in seguito). I colori sono trasmessi come segnali analogici con tensioni comprese tra 0V e 0.7V, in cui il valore più alto corrispondente al massimo di luminosità. I segnali di sincronismo invece sono dei segnali digitali a livelli TTL positivi (cioè impulsi attivi alti). In qualche caso i due segnali di sincronismo possono essere accorpati in un unico segnale "composito", che è ottenuto semplicemente dall'XOR dei due. A volte questo segnale è ulteriormente sovrapposto al segnale del verde. Dal momento che le frequenze e le bande tipiche dei segnali sono piuttosto grandi, i collegamenti prevedono un adattamento d'impedenza a 75o, e sono quindi terminati ad entrambe le estremità con un'impedenza di questo valore. Come già detto se si vuole ottenere la massima luminosità di uno dei segnali di colore occorre fornire circa 0.7V. Per fare questo utilizzando una normale uscita digitale (TTL o LVTTL) è possibile sfruttare proprio la terminazione di 75o per ottenere un effetto di partizione, o eventualmente per costruire un semplice DAC realizzato tramite un ladder resistivo (Figura 2). Se associamo un solo livello a ciascuno dei canali di colore (acceso o spento), possiamo ottenere un massimo di 8 colori (3 bit di colore). In particolare le combinazioni ottenibili sono quelle visibili in Tabella 1. Se si vuole otte-

Figura 1: Connettore VGA

ROSSO (R)	VERDE (G)	BLU (B)	COLORE
0	0	0	Nero
0	0	1	Blu
0	1	0	Verde
0	1	1	Ciano
1	0	0	Rosso
1	0	1	Magenta
1	1	0	Giallo
1	1	1	Bianco

Tabella 1

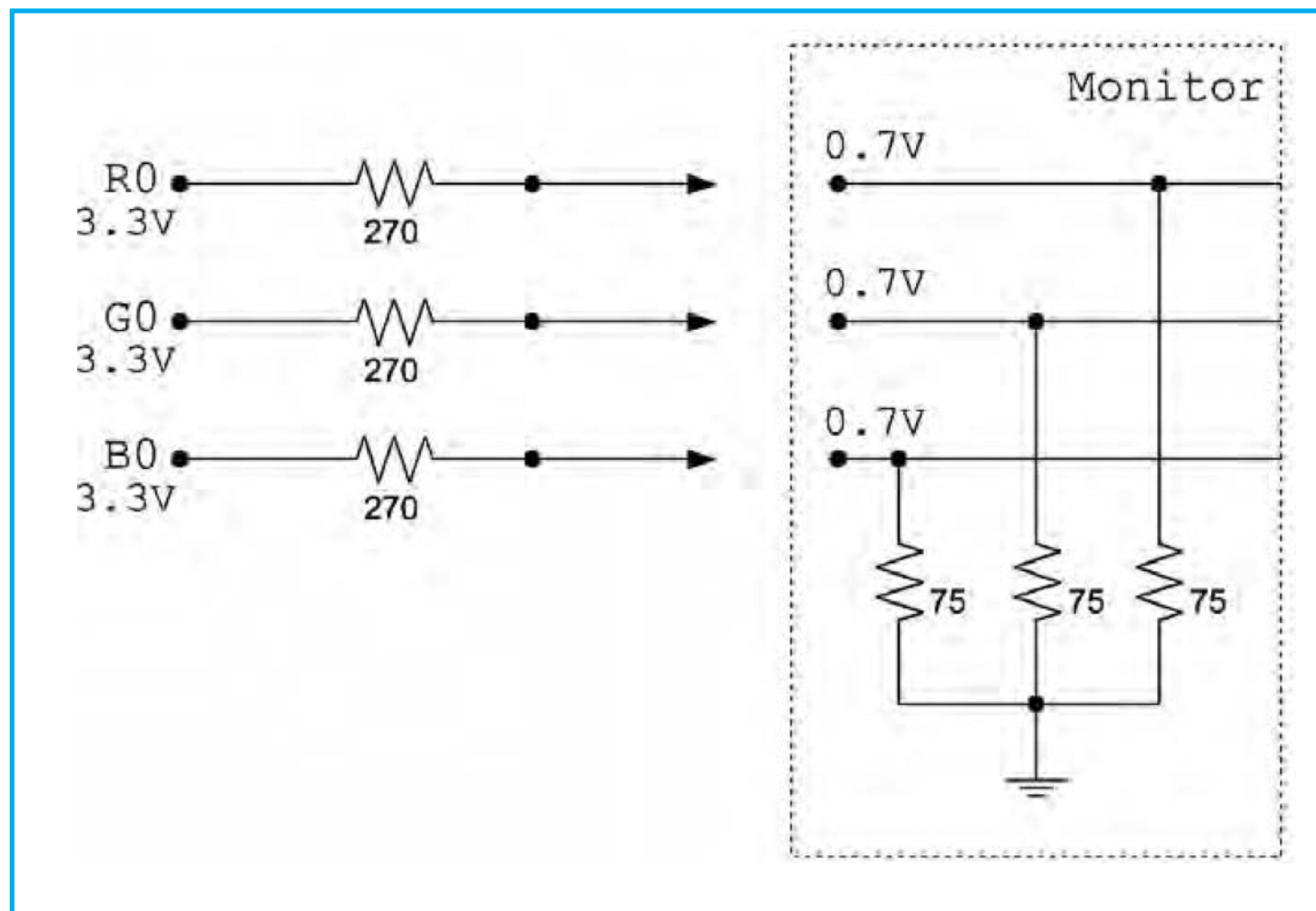



Figura 2: Rete resistiva completa per 3 bit di colore

nere un numero maggiori di colori occorre generare più di un livello per ciascun canale. Questo può essere fatto con le reti resistive mostrate in Figura 3. In questo caso

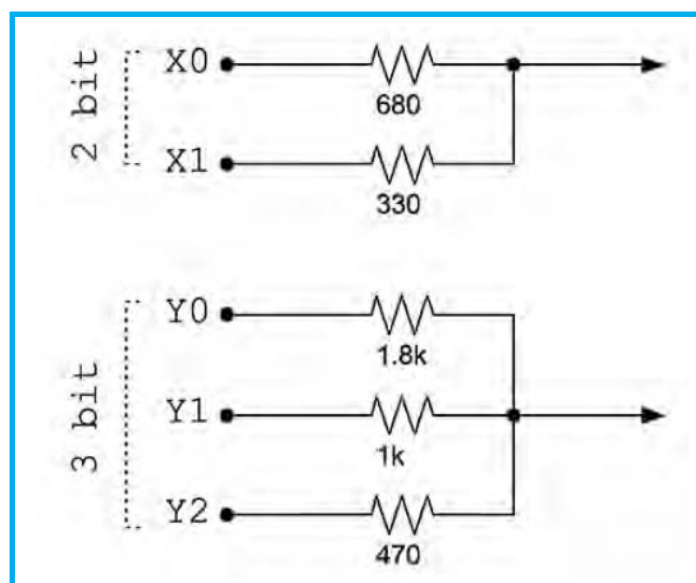



Figura 3: Reti resistive per 6 e 9 bit di colore (è mostrato solo un canale)

possono essere ottenuti rispettivamente 64 o 512 colori (6 o 9 bit di colore). Ovviamente in questi ultimi casi sarà necessario utilizzare più uscite digitali per pilotare ciascun canale. Se si vuole ottenere una risoluzione di colore ancora maggiore, come 16 o 24 bit, è conveniente usare dei DAC integrati e da 5 a 8 uscite digitali per canale. Negli esempi seguenti comunque verrà considerato per semplicità il caso di 3 bit di colore. La banda caratteristica di questi segnali può essere calcolata approssimativamente moltiplicando il numero di pixel orizzontali per quelli verticali, per il numero di quadri al secondo e per il tempo aggiuntivo impiegato per il "retrace", che conta per un fattore compreso tra 1.2 e 1.4. Ad esempio, una tipica risoluzione VGA

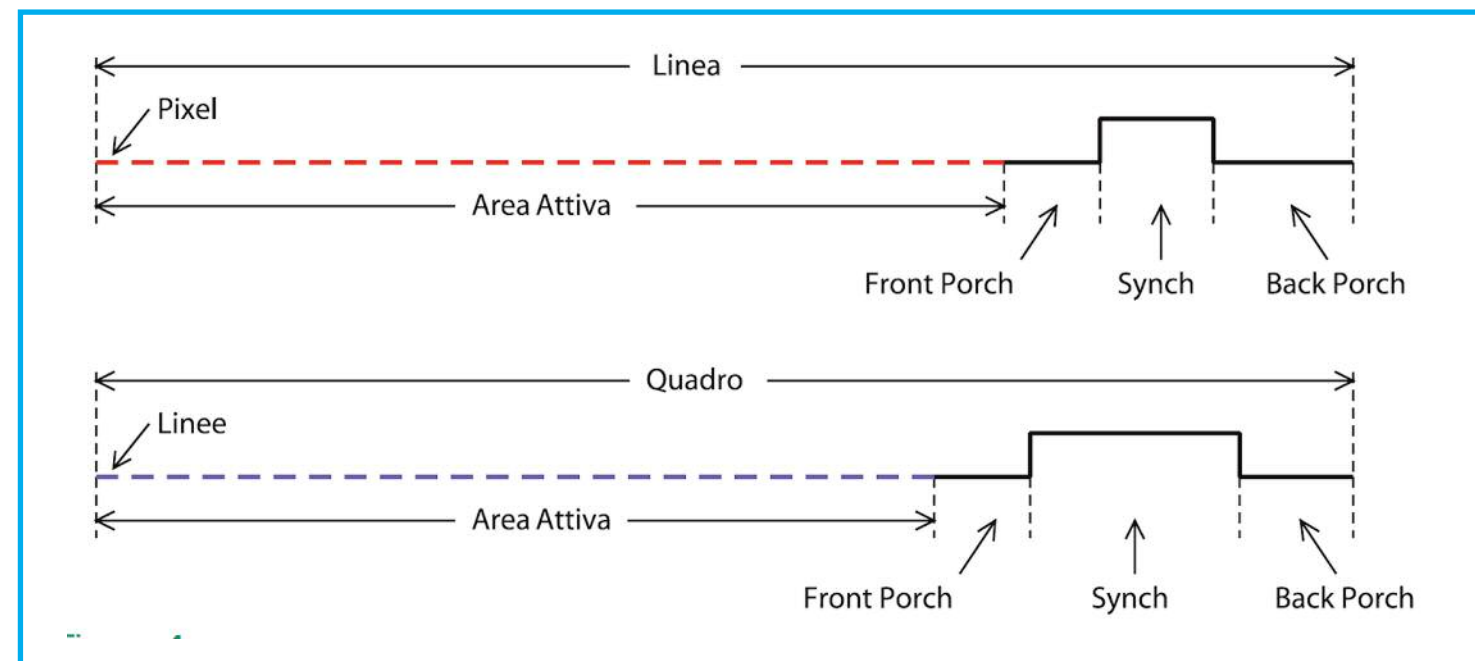



Figura 4: Temporizzazione orizzontale (in alto) e verticale (in basso)

640x480 a 60Hz richiede: 640 x 480 x 60 x 1.3 j 25MHz Questa frequenza è anche il cosiddetto "pixel clock", cioè la velocità con cui i dati di colore relativi ad ogni pixel devono essere presentati alle uscite. Si intuisce da questa considerazione il perché la generazione di un segnale VGA richieda hardware dedicato, o comunque molto veloce come FPGA o CPLD.

TEMPORIZZAZIONI

Le temporizzazioni orizzontali e verticali di un segnale VGA sono schematizzate in Figura 4. Si può notare come la temporizzazione orizzontale (H) e quella verticale (V) siano abbastanza simili, anche se su scale temporali diverse (il segnale verticale è molto più lento). In entrambi i casi è presente una "zona attiva", in cui sono presentati alle uscite i valori di colore associati a ciascun pixel, o le varie linee. Terminata la zona attiva si ha un intervallo chiamato "piedistallo anteriore" ("Front Porch"), poi l'impulso di sincronismo, ed un "piedistallo posteriore" ("Back Porch"). Dall'inizio del piedistallo anteriore e fino al nuovo inizio dell'area attiva i segnali di colore devono essere spenti (portati a 0V). Questo è necessario perché in questo tempo, chiamato intervallo di "Blank orizzontale", il pennello elettronico (nei monitor CRT) deve spostarsi dal bordo destro a quello sinistro per ricominciare la scansione della linea successiva.
Molti dispositivi video, anche non basati su CRT, non funzionano correttamente se non si spengono i segnali di colore durante i blank. Per le temporizzazioni verticali si ha una situazione simile, solo che in questo caso si contano le linee tracciate e non i pixel. Dopo un certo numero di linee (dipendente dalla risoluzione utilizzata) si avrà un intervallo di blank verticale, che comprenderà i due piedistalli e l'impulso di sincronismo. Dal momento che i tempi verticali si contano in linee e non in pixel, risultano molto più lenti di quelli orizzontali. La Figura 5 fornisce una visione d'insieme,

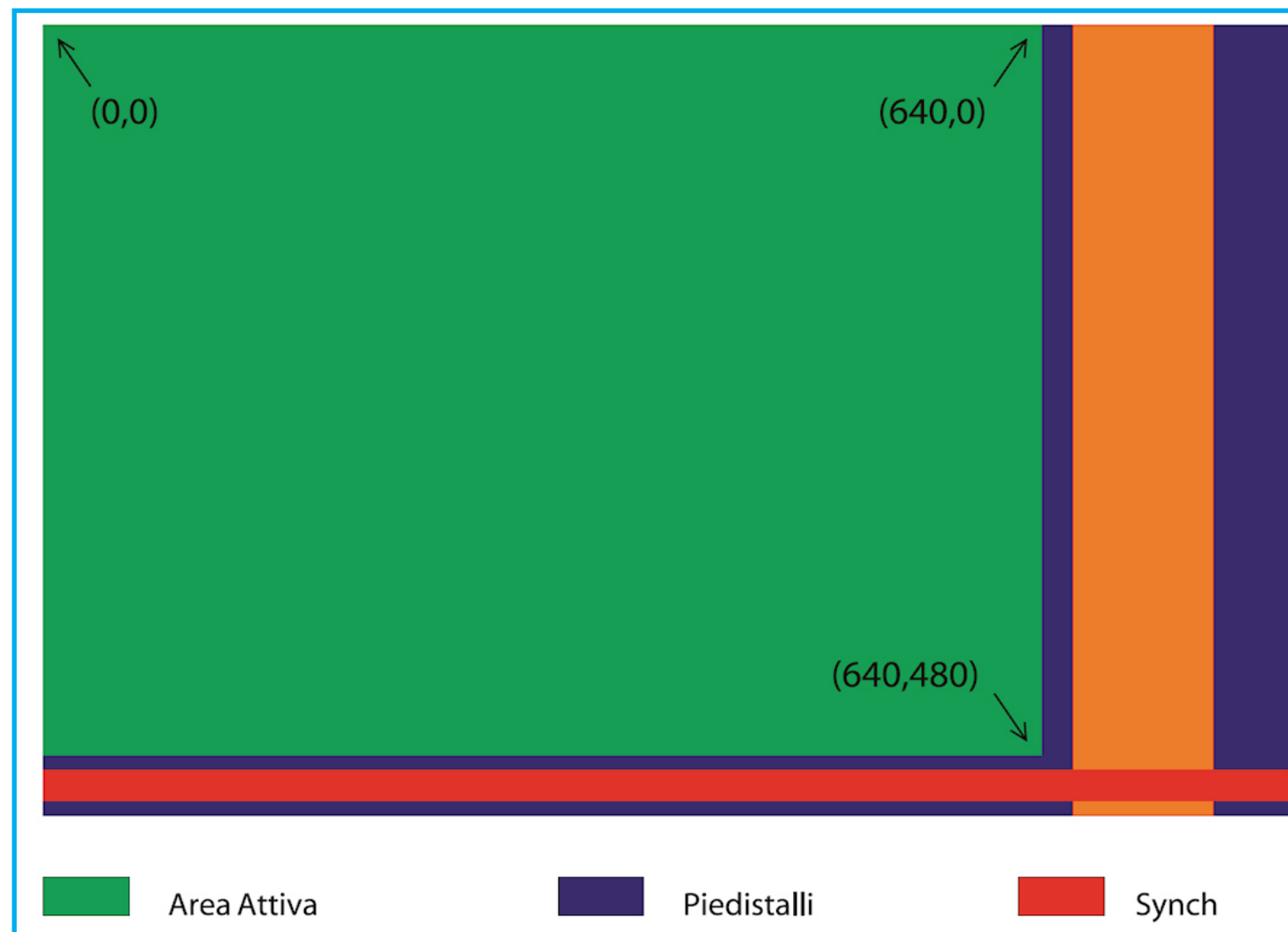



Figura 5: Temporizzazioni H e V sovrapposte ed in scala

in scala, dei tempi e della successione degli eventi (se si tiene conto di come avviene la scansione). Per ottenere diverse risoluzioni è sufficiente scegliere la velocità con cui si forniscono i dati colore relativi ai pixel (il pixel clock), e la dimensione dei vari intervalli. In Tabella 2 sono riassunti i valori relativi alle risoluzioni più comuni. I dispositivi video sono in grado riconoscere i tempi utilizzati e di agganciarsi a questi. Molti monitor CRT consentono anche una certa flessibilità, e sono in grado di agganciarsi anche se i tempi dei segnali forniti in ingresso non risultano particolarmente precisi. Altri dispositivi più recenti (ad esempio i videoproiettori), anche per motivi tecnologici, richiedono una precisione maggiore, e potrebbero non funzionare correttamente se non si rispettano perfettamente i tempi indicati o supportati.

GENERAZIONE DEI SINCRONISMI

Come già visto è necessario che i segnali di colore e sincronismo siano forniti con le opportune temporizzazioni. In questo paragrafo sarà analizzata la generazione dei sincronismi, che tra l'altro risulta indipendente dal contenuto dell'immagine.
Dal momento che si tratta di segnali periodici con tempi caratteristici multipli del pixel clock, il modo più semplice per generarli consiste nel partire proprio dal pixel clock ed utilizzare dei contatori. In base al valore raggiunto da questi sarà possibile

FORMATO	PIXEL CLOCK (MHZ)	ORIZZONTALE (PIXEL)					VERTICALE (LINEE)				
		AREA ATTIVA	FRONT PORCH	SYNCH	BACK PORCH	TOT.	AREA ATTIVA	FRONT PORCH	SYNCH	BACK PORCH	TOT.
640x480, 60Hz	25.175	640	16	96	48	800	480	11	2	31	524
640x480, 72Hz	31.500	640	24	40	128	832	480	9	3	28	520
640x480, 75Hz	31.500	640	16	96	48	800	480	11	2	32	525
640x480, 85Hz	36.000	640	32	48	112	832	480	1	3	25	509
800x600, 56Hz	38.100	800	32	128	128	1088	600	1	4	14	619
800x600, 60Hz	40.000	800	40	128	88	1056	600	1	4	23	628
800x600, 72Hz	50.000	800	56	120	64	1040	600	37	6	23	666
800x600, 75Hz	49.500	800	16	80	160	1056	600	1	2	21	624
800x600, 85Hz	56.250	800	32	64	152	1048	600	1	3	27	631
1024x768, 60Hz	65.000	1024	24	136	160	1344	768	3	6	29	806
1024x768, 70Hz	75.000	1024	24	136	144	1328	768	3	6	29	806
1024x768, 75Hz	78.750	1024	16	96	176	1312	768	1	3	28	800
1024x768, 85Hz	94.500	1024	48	96	208	1376	768	1	3	36	808

Tabella 2

distinguere quale parte del segnale generare.
Si avrà quindi in sequenza la zona attiva in cui verranno aggiornati i segnali di colore, il front porch da cui inizierà l'intervallo di blank, l'impulso di sincronismo ed il back porch. Alla fine del back porch il contatore verrà resettato ed il ciclo inizierà daccapo.
Per i segnali verticali si utilizza la stessa tecnica, ma il contatore non sarà incrementato dal pixel clock, ma da ogni fine di linea orizzontale.
Il codice VHDL che implementa la generazione dei segnali di sincronismo è riportato nel Listato 1. Il codice implementa i due contatori (HCNT e VCNT), gestiti in due processi separati, che sono utilizzati per generare i segnali di sincronismo e di blank. Questi segnali costituiscono le uscite del modulo VHDL, che possono essere utilizzati per coordinare la generazione dei segnali di colore da parte di un modulo esterno. Per cambiare la risoluzione è necessario modificare le specifiche sulle durate degli intervalli, che sono indicate nelle costanti (i valori sono gli stessi indicati in Tabella 2, in esadecimale), e fornire il pixel clock corretto.

GENERAZIONE DI UNA IMMAGINE

È possibile generare l'immagine (cioè la successione di valori da fornire come segnali colore) in diversi modi. Sono comuni tre metodi: a partire dalle coordinate attuali (X,Y), utile per generare pattern o semplici disegni o animazioni; a partire da un indirizzo lineare, adatto a visualizzare bitmap memorizzate in una memoria; a partire da riga, colonna e scanline, adatto a visualizzare testo o grafica a tasselli ("tile"). Ciascuno di questi metodi può essere implementato facilmente a partire dai segnali forniti dal modulo di sincronismo. Ci si concentrerà qui sul metodo più comune, semplice e versatile, cioè quello basato su indirizzi lineari.
È sufficiente in questo caso utilizzare un contatore che si incrementa ad ogni ciclo del pixel clock quando il segnale di blank non è attivo. Il valore del contatore può essere utilizzato per indirizzare una memoria RAM statica che contiene proprio il valore dei segnali colore per ciascun pixel. La quantità di memoria richiesta (in bit) può essere calcolata moltiplicando il numero di pixel dell'immagine per il numero di bit necessari per rappresentare i colori. In effetti storicamente il principale ostacolo a risoluzioni alte è stata proprio la mancanza o l'alto costo della memoria. Per ovviare a questo limite fino ad una decina di anni fa (soprattutto nelle macchine da gioco) si ricorreva ad un semplice trucco: si alternavano linee dell'immagine con linee vuote (nere), questo permetteva di dimezzare la quantità di memoria richiesta.
Riguardo alla memoria bisogna considerare anche un altro problema: la velocità di accesso. Ad una risoluzione di 800x600 occorre leggere i dati con una frequenza di 50MHz, e se i dati richiedono più byte occorre eseguire più letture. Occorrono quindi memorie capaci di tempi di accesso molto piccoli (inferiori a 10ns) o con sufficiente larghezza di parola (16 o 32 bit). La soluzione qui utilizzata si basa sull'impiego di una comune memoria SRAM da 4Mbit organizzata in 256K word da 16 bit (es. Hitachi HM6216255 o equivalenti). L'immagine sarà memorizzata nella SRAM come successione ordinata di pixel. Ciascuna word codifica 4 pixel disposti come segue: -BGR -BGR -BGR -BGR, in cui il bit meno significativo si riferisce al primo pixel (nell'ordine di scansione). Saranno necessarie: 640 x 480 x 4 / 16 = 76800 word. Il codice VHDL che richiama il modulo di generazione dei sincronismi e che genera i segnali per accedere alla memoria è riportato nel Listato 2.
Il codice non fa altro che generare gli indirizzi utilizzando un contatore incrementato dal pixel clock e bloccato dal segnale di blank (si è supposto di utilizzare un clock in ingresso di 50MHz, quindi è stata prevista una divisione per 2 della frequenza). Gli indirizzi forniti alla memoria devono essere divisi per 4, in modo da tenere in conto il fatto che vengono letti 16 bit alla volta. I dati letti vengono quindi forniti sequenzialmente in uscita (multiplexati usando i due bit meno significativi del contatore). Si può notare anche che i segnali di colore sono azzerati in corrispondenza dell'intervallo di blank, mentre il contatore degli indirizzi è azzerato in corrispondenza dell'impulso di sincronismo verticale.

SCHEMA ELETTRICO

Lo schema dei collegamenti relativi all'interfaccia video è mostrato in Figura 6. Lo schema non è dettagliato perché il codice e le tecniche proposte possono essere implementate su una notevole quantità di dispositivi (FPGA, CPLD, PLD, logica discreta, etc.). Il codice VHDL presentato è stato testato su una FPGA Xilinx Spartan3, e la sua implementazione ha richiesto cir-

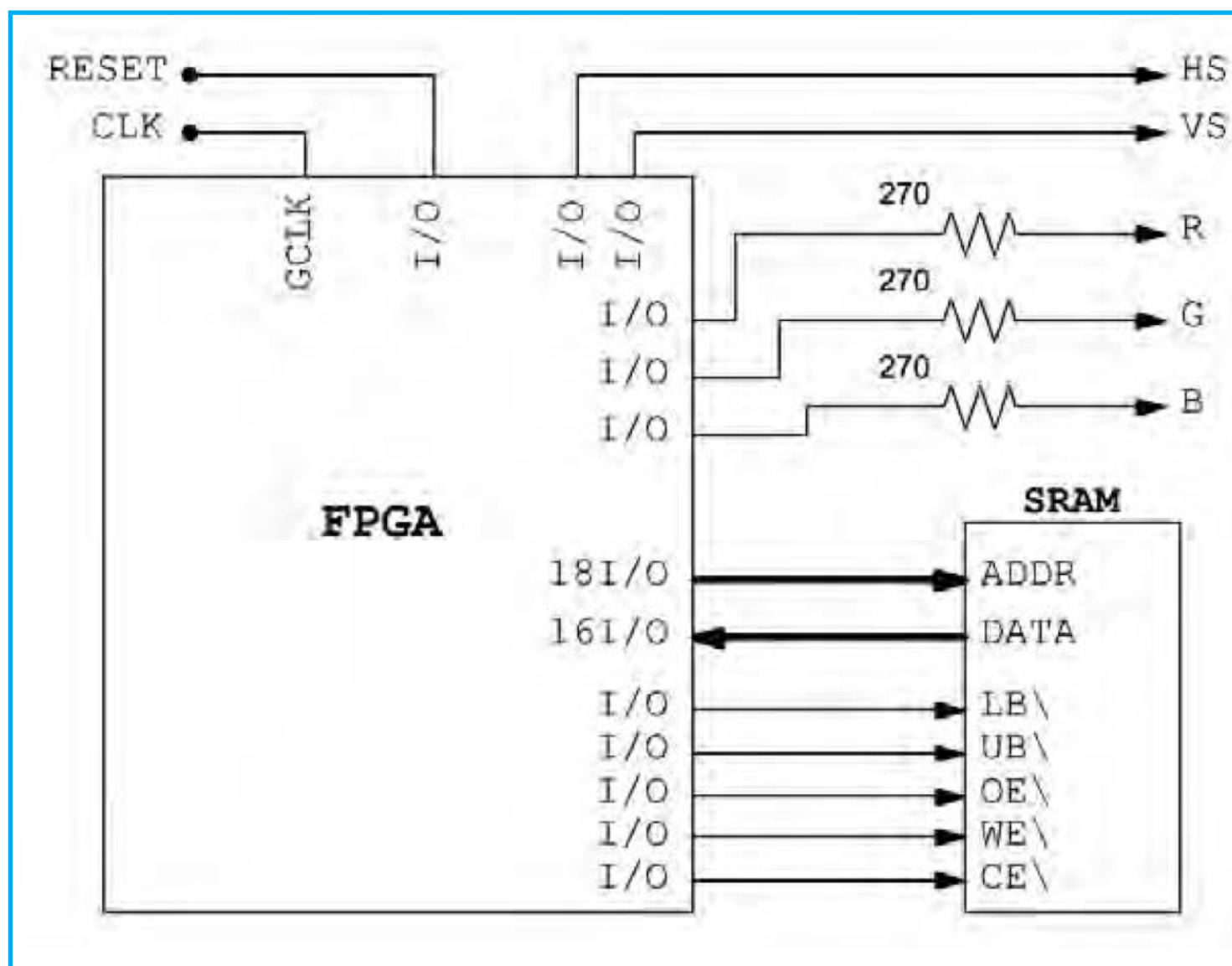



Figura 6: Schema elettrico per la connessione della FPGA

ca 50 slices. È stato utilizzato un clock di 25MHz per testare la risoluzione 640x480 a 60Hz ed uno a 50MHz per la 800x600 pixel a 72Hz. Nonostante la prima frequenza sia un po' inferiore a quella prevista, tutti i dispositivi video provati sono stati in grado visualizzare correttamente il segnale, in entrambe le risoluzioni. Va notato che il codice presentato non permette di scrivere i dati nella memoria, perché questo aspetto dipende strettamente dall'applicazione per cui si sta sviluppando l'interfaccia. In generale si possono utilizzare due tecniche: o si utilizza una memoria a doppia porta, di cui una utilizzata dal controller (es. FPGA) per la lettura, e l'altra lasciata libera per la scrittura da parte di un dispositivo esterno qualsiasi (ad esempio un microcontrollore); oppure si può utilizzare una memoria a singola porta, e si implementa un arbitro nel controller, in modo da potere gestire la lettura e la scrittura in modo che queste operazioni non interferiscano. Il primo metodo è molto semplice e consente un'elevata velocità di aggiornamento dei dati, il secondo consente la scrittura praticamente soltanto durante gli intervalli di blank, ma nonostante questo in pratica è il più utilizzato grazie al suo basso costo.

LISTATO 1

```
— Formato: VGA a 640x480 - 60Hz – 25MHz
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity VGA_Synch is
Port (HS : out std_logic;
VS : out std_logic;
BLANK : out std_logic;
CLK : in std_logic;
RESET : in std_logic);
end VGA_Synch;
architecture RTL of VGA_Synch is
— * Costanti *
constant RESET_ACTIVE : std_logic := ‘1’; — Valore reset attivo
— * 640x480, 25MHz *
constant H_TOTAL : std_logic_vector(11 downto 0) := x”320”;
constant H_ACTIVE : std_logic_vector(11 downto 0) := x”280”;
constant H_FRONT_PORCH : std_logic_vector(11 downto 0) := x”010”;
constant H_BACK_PORCH : std_logic_vector(11 downto 0) := x”030”;
constant V_TOTAL : std_logic_vector(11 downto 0) := x”20C”;
constant V_ACTIVE : std_logic_vector(11 downto 0) := x”1E0”;
constant V_FRONT_PORCH : std_logic_vector(11 downto 0) := x”00B”;
constant V_BACK_PORCH : std_logic_vector(11 downto 0) := x”01F”;
— * Segnali *
signal HCNT : std_logic_vector(11 downto 0);
signal VCNT : std_logic_vector(11 downto 0);
signal HBLANK : std_logic;
signal VBLANK : std_logic;
signal CBLANK : std_logic;
begin
— *** Sincronismo orizzontale ***
process(CLK, RESET)
begin
if RESET = RESET_ACTIVE then
HCNT <= (others => ‘0’);
HS <= ‘0’;
HBLANK <= ‘0’;
elsif(CLK’event and CLK=’1’) then
— Contatore orizzontale
if HCNT=(H_TOTAL-1) then
HCNT <= (others => ‘0’);
HBLANK <= ‘0’;
else
HCNT <= HCNT + 1;
end if;
— Generazione impulso di sincronismo
if HCNT=(H_ACTIVE+H_FRONT_PORCH-1) then
HS <= ‘1’;
elsif HCNT=(H_TOTAL-H_BACK_PORCH-1) then
HS <= ‘0’;
end if;
— Blank orizzontale
if HCNT=(H_ACTIVE-1) then
HBLANK <= ‘1’;
end if;
end if;
end process;
— *** Sincronismo verticale ***
process(CLK, RESET)
begin
if RESET = RESET_ACTIVE then
VCNT <= (others => ‘0’);
VS <= ‘0’;
VBLANK <= ‘0’;
elsif(CLK’event and CLK=’1’ and HCNT=(H_TOTAL-1))
then
— Contatore orizzontale
if VCNT=(V_TOTAL-1) then
VCNT <= (others => ‘0’);
VBLANK <= ‘0’;
else
VCNT <= VCNT + 1;
end if;
— Generazione impulso di sincronismo
if VCNT=(V_ACTIVE+V_FRONT_PORCH-1) then
VS <= ‘1’;
elsif VCNT=(V_TOTAL-V_BACK_PORCH-1) then
VS <= ‘0’;
end if;
— Blank verticale
if VCNT=(V_ACTIVE-1) then
VBLANK <= ‘1’;
end if;
end if;
end process;
CBLANK <= HBLANK or VBLANK;
BLANK <= CBLANK;
end RTL;
```

LISTATO 2

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity VideoGen is
Port (DATA : in std_logic_vector(15 downto 0);
ADDR : out std_logic_vector(17 downto 0);
CE : out std_logic;
OE : out std_logic;
WE : out std_logic;
LB : out std_logic;
UB : out std_logic;
R : out std_logic;
G : out std_logic;
B : out std_logic;
HS : out std_logic;
VS : out std_logic;
CLK : in std_logic;
RESET : in std_logic);
end VideoGen;
architecture Behavioral of VideoGen is
signal BLANK : std_logic;
signal RED : std_logic;
signal GREEN : std_logic;
signal BLUE : std_logic;
signal HSI : std_logic;
signal VSI : std_logic;
signal PADDR : std_logic_vector(19 downto 0);
signal MDATA : std_logic_vector(15 downto 0);
signal CLKDIV : std_logic;
signal MCE : std_logic;
component VGA_Synch
Port (HS : out std_logic;
VS : out std_logic;
BLANK : out std_logic;
CLK : in std_logic;
RESET : in std_logic);
end component ;
begin
process(CLK, RESET)
begin
if RESET='1' then
CLKDIV <= '0';
MDATA <= (others => '0');
PADDR <= (others => '0');
MCE <= '1';
elsif(CLK'event and CLK='1') then
CLKDIV <= not CLKDIV;
if (BLANK='0' and CLKDIV='1') then
PADDR <= PADDR + 1;
end if;
if VSI='1' then
PADDR <= (others => '0');
end if;
MCE <= '1';
if PADDR(1 downto 0)="00" then
MCE <= '0';
MDATA <= DATA;
end if;
end if;
end process;
LB <= '0';
UB <= '0';
OE <= '0';
WE <= '1';
CE <= MCE;
ADDR <= PADDR(19 downto 2);
RED <= MDATA(0) when PADDR(1 downto 0)="00"
else
MDATA(4) when PADDR(1 downto 0)="01"
else
MDATA(8) when PADDR(1 downto 0)="10"
else
MDATA(12);
GREEN <= MDATA(1) when PADDR(1 downto 0)="00"
else
MDATA(5) when PADDR(1 downto 0)="01"
else
MDATA(9) when PADDR(1 downto 0)="10"
else
MDATA(13);
BLUE <= MDATA(2) when PADDR(1 downto 0)="00"
else
MDATA(6) when PADDR(1 downto 0)="01"
else
MDATA(10) when PADDR(1 downto 0)="10"
else
MDATA(14);
R <= RED and (not BLANK);
G <= GREEN and (not BLANK);
B <= BLUE and (not BLANK);
HS <= HSI;
VS <= VSI;
SYNCH1: VGA_Synch Port Map
(HS => HSI,
VS => VSI,
BLANK => BLANK,
CLK => CLKDIV,
RESET => RESET);
end Behavioral;
```

VARIALUCE A MICROCONTROLLORE

Un circuito, basato su PIC, per variare l'intensità luminosa di lampade a incandescenza o alogene di potenza fino a 300W

Con questo dimmer potrete controllare 4 lampade alogene da 50 Watt ciascuna, e funziona anche in impianti che prevedono un doppio deviatore (come ad esempio nel caso delle luci delle scale).

LO SCHEMA

Il regolatore di luce a micro, regola l'intensità luminosa controllando la fase del segnale in uscita: non c'e da stupirsi quindi se il triac TRI1 farà tutto il lavoro più pesante. Prima pero che il triac cominci a funzionare, devono essere considerati alcuni valori; cominciando dall'inizio: la tensione alternata di rete (vedi **figura 1**), la bobina L1 e il condensatore C2 formano il classico filtro per prevenire i disturbi generati dal circuito di controllo di fase, probabile colpevole per eventuali interferenze al-la linea d'uscita. F1 e il fusibile in miniatura da 1.6 A (ritardato) che garantisce la protezione del circuito. R1, R2, C1, B1 e D1 convertono i 230V in ingresso, in una tensione adeguata al campo di lavoro di IC1, un regolatore 78L05 . Per prevenire eventuali cadute di tensione, si utilizzano due resistenze da 470Ω, invece che un'unica da 1 kΩ. Connettendo i due resistori da 470Ω (R1 e R2) in serie il rischio di deterioramento elettrico si riduce, in quanto la caduta in ogni resistenza e la meta di quella totale.

Normalmente una tensione di alimentazione pari a 16V per il 78L05 e troppo elevata, ma considerando che l'energia deve essere immagazzinata in C3 e C4 in un tempo molto breve, si e scelto di lavorare con la tensione più alta possibile. L'energia immagazzinata in C3 e C4 deve alimentare il microcontrollore per il tempo necessario ad attivare la modalità impostazione; infatti la "setting mode" viene eseguita spegnendo la lampada (e quindi il circuito) e riaccendendola immediatamente (vedere più avanti per la descrizione completa d'utilizzo).

VINCOLI DI SPAZIO

Le necessita imponevano di alloggiare il circuito all'interno di un contenitore alto solamente due centimetri, obbligando quindi il montaggio di C1 e C2 sul fi anco, e non verticalmente come di consueto (vedi foto). Per lo stesso motivo l'alimentazione si avvale di due piccoli condensatori elettrolitici (C3 e C4), anziché utilizzarne un unico di dimensioni più elevate.

Nel caso si avessero a disposizione più di due centimetri, gli accorgimenti appena descritti non sono realmente necessari, in ogni modo a causa di L1 resta poco spazio per C1 e C2. Un problema derivante da questo modello d'alimentazione riguarda il microcontrollore, potenzialmente connesso alla corrente alternata. Sta di fatto che un trasformatore risultava troppo ingombrante per essere inserito all'interno di un case cosi piccolo, quindi si e optato per il sistema appena descritto.

CUORE DEL CIRCUITO

Il nucleo del circuito e formato da un microcontrollore PIC12F625, mentre il rilevamento dello zero-crossing e realizzato me-

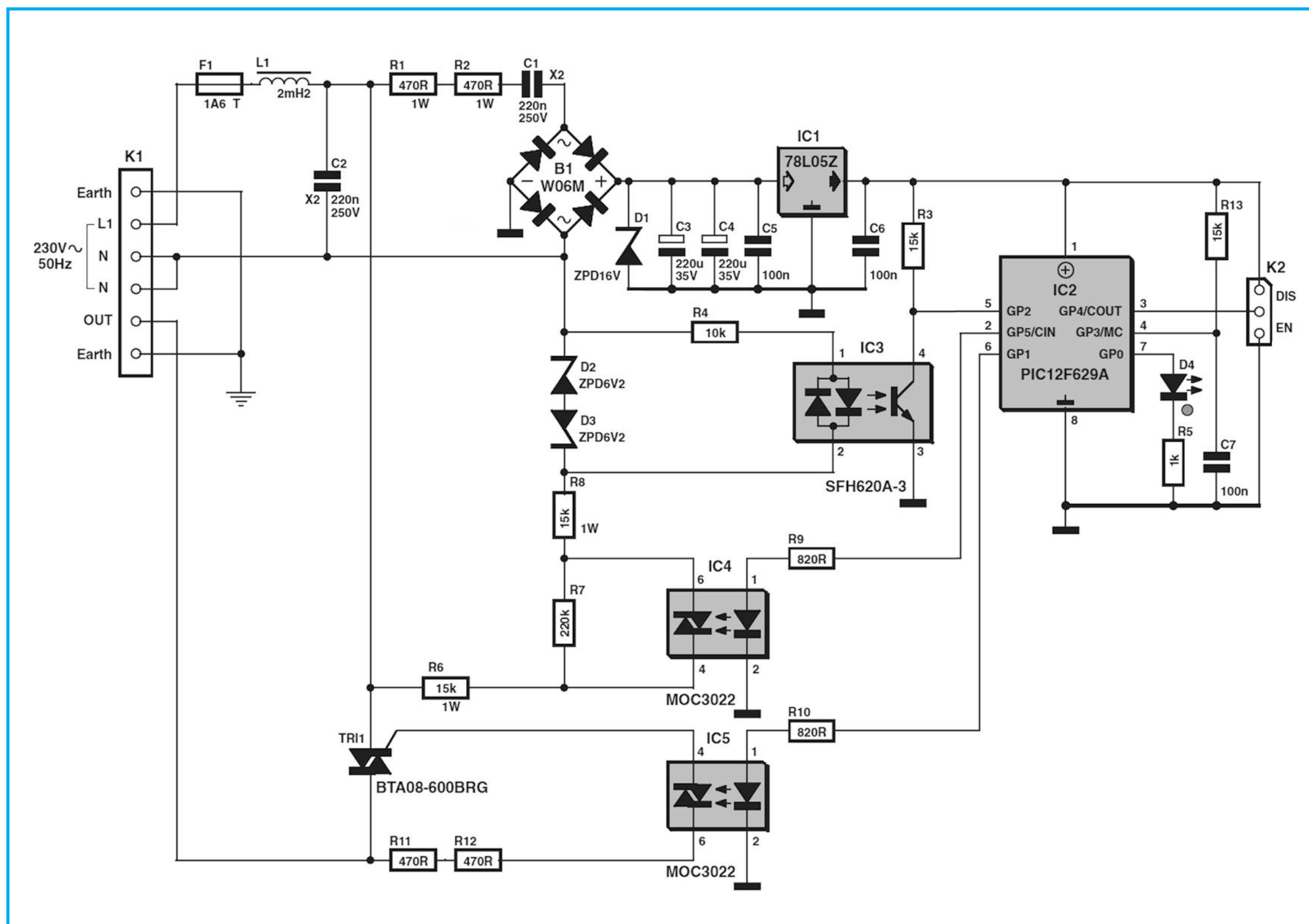



Figura 1. Schema elettrico del dimmer

diante l'integrato IC3: infatti la tensione al-l'uscita e più alta di 2.5V con una durata di 2ms. D2 e D3 limitano la tensione ai capi dei led di IC3. Utilizzando gli impulsi derivanti da IC3, il microcontrollore sa quando l'onda sinusoidale passa per lo zero attivando, a seconda della posizione del dimmer, il triac. La resistenza R3 serve come resistore di pull-up.

Si sarebbero potute utilizzare le resistenze interne al microcontrollore, ma per risparmiare nell'assorbimento se ne preferito l'inserimento esterno all'integrato: tutto ciò a causa dell'impossibilità di attivare i pull-up solamente negli ingressi desiderati. Si e notato che utilizzando una singola resistenza di pull-up esterna, la quantità di tempo in cui il circuito riesce ad operare senza alimentazione e notevole(importante quando si deve impostare la posizione del dimmer). D4 si attiva quando si entra in "setting mode". Nel caso l'indicazione della "setting mode" da parte del LED non fosse necessaria si può liberalmente decidere di omettere D4 e R5, aumentando notevolmente la quantità di tempo in cui il circuito riesce ad operare senza alimentazione. Ciò è causato dal fatto che C3 e C4 vengono scaricati più lentamente: niente viene dissipato da D4.

CALORE

R8 e R6 saranno leggermente calde, considerato che devono supportare tutta la tensione di rete. Ciò perché, con l'aiuto di IC4, R7 e R9, alla corrente sara permesso circolare solamente attraverso R8 e R6 prima dello zero-crossing. Cosi facendo viene determinato con accuratezza l'esatto punto di accensione del triac, minimizzando anche la corrente che attraversa R6 e R8.

Se ci fosse stato abbastanza spazio per raggiungere una corretta dissipazione termica di R6, R8 e IC4, allora R7 e R9 non avrebbero avuto ragione d'esistere (connettendo il pin 2 di IC2 direttamente al nodo tra R6 e R8). R6 e R8 dissiperanno 0.9W ciascuna, quindi resistenze da 1W lavoreranno perfettamente. Senza IC4 che porta il livello del calore sotto i limiti massimi, ci sarebbe stato un possibile rischio d'incendio.

Non appena IC2 rileva il primo zero-crossing, il controller attraverso IC4 azzera la corrente che attraversa R6 e R8, finchè non si raggiunge il successivo istante di alimentazione. Poco prima che accada ciò, il triac IC4 si attiva. Come risultato si ottiene che i resistori R6 e R8 restano freddi. R7 serve a far in modo che IC4 si

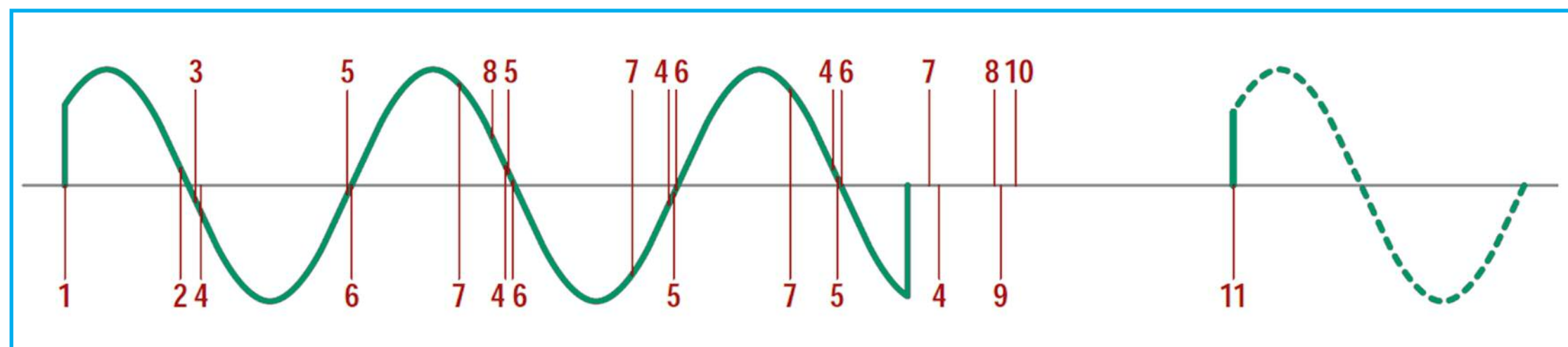



Figura 2. Il diagramma mostra i punti essenziali al programma durante l'andamento temporale della corrente alternata.

spenga dopo il punto di zero-crossing. IC5 viene utilizzato per interfacciare il triac TRI1 con il microcontrollore. Il progetto non utilizza dispositivi con case TO-220, a causa della distanza ridotta dei pad rispetto agli altri stampati. Infatti in questo caso le connessioni degli integrati son distanti 1.25 mm, per prevenire corto circuiti.

IL SOFTWARE

Il programma si suddivide in tre parti. La prima parte contiene la gestione del dimmer, la seconda (centrale) il codice necessario ad utilizzare la modalita setting, mentre l'ultima la gestione della memoria EEPROM. Dopo aver fornito alimentazione, i registri vengono inizializzati e il valore del dimmer letto dalla memoria. Il microcontrollore attende per due intervalli di zero-crossing (passo 4), in modo che nessun rimbalzo dell'interruttore influisca sul funzionamento. La lista qui sotto corrisponde con i numeri nella sinusoide in **figura 2** e le routine nel codice:

1. Connettere la rete elettrica, il microcontrollore avvierà la procedura di inizializzazione
2. Pretest0; il PIC attende lo zero-crossing.
3. Pretest1; il micro attende che passi lo zero-crossing, il pin 5 e nuovamente basso
4. Main; attende il successivo zero-crossing o che sia tolta la corrente.
5. MainWaitToSwitchTriacOnSetup; Resetta il timer interno.
6. MainWaitToSwitchTriacOn; Attende che il timer raggiunga il valore desiderato per sincronizzare il TRIAC.
7. Il Triac è sincronizzato, il piedino 6 del pic va a livello alto rapidamente.
8. MainWaitToSwitchOnZCrossDetect; viene attivato il rilevatore di zero-crossing perché il pin 2 passa a stato alto. Il programma salta indietro al punto 4.
9. Programma; il PIC entra in setting mode e verifica la posizione del jumper. A seconda della posizione, il sistema salta al passo uno o continua con l'esecuzione.
10. ShowProgramMode; il LED si accende, e il TRIAC comincia a condurre.
11. Seguendo la descrizione precedente si può facilmente determinare l'operazione che il programma gestisce quando si trova in modalita setting. Il codice per la lettura\scrittura nella EEPROM deriva direttamente dal datasheet della Microchip.

OPERAZIONI

Il dispositivo può essere configurato in "setting mode". Questa modalità viene attivata accendendo, spegnendo e accendendo nuovamente la lampada (con un intervallo massimo di un secondo tra una fase e l'altra).
La lampada si accenderà inizialmente alla massima potenza per circa un secondo, per poi essere regolata attraverso una scale che va dallo 0% al 100%. Quando viene raggiunta la massima potenza, il ciclo si ripete ripartendo dallo 0%. Non appena viene raggiunta la quantità di luce desiderata, si dovrà solamente spegnere l'interruttore per circa due secondi (o finche il led non si spegne), salvando in questo modo il valore desiderato. Non appena verrà accesa nuovamente la luce, la lampada raggiungerà il livello di luminosità pre-impostato.
Con il ponticello K2 si può decidere di disabilitare la modalità setting, in modo che il dimmer utilizzi l'ultima configurazione di luminosità memorizzata.
Per fare ciò utilizzare il jumper per connettere il pin 3 del microcontrollore direttamente ai +5V. Se si considera il periodo on-off-on troppo lungo per avviare la modalità d'impostazione, basterà modificare il valore dei condensatori C3 e C4 al di sotto dei 100 μF; d'altro canto se il tempo sembra troppo breve basterà effettuarne la sostituzione con due capacita più elevate.

Elenco componenti	
R1,R2	470Ω 1W
R3, R13	15kΩ
R4	10kΩ
R5	1kΩ
R6,R8	15kΩ 1W
R7	220kΩ
R9,R10	820Ω
11,R12	470Ω
C1,C2	220nF 250V
C3,C4	220 F 35V
C5,C6,C7	100nF
B1	W06M, ponte a diodi, 1.5A, 600Vpiv
D1	16V 0.5W zener
D2,D3	6.2V 0.5W zener
D4	LED, verde, 3mm
IC1	78L05 (TO-92 case)
IC2	PIC12F629A
IC3	SFH620A-3 fotoaccoppiatore
IC4,IC5	MOC3022 fotoaccoppiatore
TRI1	BTA08-600BRG, triac, 8A, 600V,
	TO-220AB
F1	fusibile 1.6 A
L1	2.2mH bobina

AIR MOUSE

Il cuore del progetto è un accelerometro MEMS che capta i vostri movimenti e muove un puntatore in base ad essi

Prima di entrare nei dettagli del progetto è utile fare una introduzione sugli accelerometri ed il loro funzionamento. Un accelerometro è un dispositivo inerziale per la misura delle accelerazioni lineari. Un dispositivo inerziale è in grado di fornire una misurazione senza bisogno di un riferimento esterno, a parte le condizioni iniziali fornite allo start-up, come ad esempio la posizione iniziale - ma solo se questo è richiesto dalla particolare applicazione. Una categoria particolare è quella degli accelerometri MEMS (Micro Electro-Mechanical Systems), dei dispositivi che integrano in un solo chip di silicio sia le parti meccaniche che i circuiti elettronici di condizionamento e controllo. Tra i maggiori produttori di accelerometri ed altri sensori MEMS troviamo Analog Devices , Freescale Semiconductor ed ST Microelectronics . Gli accelerometri ed altri tipi di sensori basati sulla tecnologia MEMS, come i giroscopi MEMS - capaci di rilevare velocità angolari - stanno conquistando campi di applicazioni sempre più ampi che vanno dagli apparati consumer - come smartphone evoluti e console di gioco - agli apparati di controllo industriale e biomedicali. La crescita del settore è pari a circa il 20% annuo e si prevede che per il 2010 praticamente tutti i nuovi dispositivi mobili saranno dotati di un accelerometro (fonte: ST Microelectronics). Gli accelerometri MEMS permettono di eseguire diversi tipi di misurazioni, sia direttamente che tramite una successiva elaborazione.

- Misurazione della velocità e della posizione:
  La velocità è ricavata come integrale dell'accelerazione: il risultato è a meno di una costante corrispondente alla velocità iniziale. La posizione è ricavata come integrale doppio dell'accelerazione, ovvero integrando la velocità se calcolata in precedenza; in questo caso è necessario conoscere le condizioni iniziali di velocità e posizione.
- Misurazione di vibrazioni e di impatti (shock).
- Misurazione dell'orientamento: poiché gli accelerometri sono sensibili alla forza di gravità, possono essere utilizzati per rilevare l'inclinazione (tilt) del dispositivo. I campi applicativi dei MEMS sono numerosi e probabilmente se ne aggiungeranno in futuro molti altri. Attualmente le applicazioni più diffuse riguardano:
- dispositivi di input per Realtà Virtuale e console di gioco;
- apparati mobili come PDA, smartphone,
- lettori multimediali;
- antifurti per laptop e simili;
- dispositivi anti caduta per HDD;
- motion sensing in Robotica;
- apparati biomedicali;
- rilevazione di urti nei sistemi Airbag.
- Controllo delle condizioni di macchine elettromeccaniche.

CARATTERISTICHE E UTILIZZI DEGLI ACCELEROMETRI MEMS

Un MEMS è un dispositivo elettromeccanico realizzato con le stesse tecnologie usate per la fabbricazione dei chip: fotoli-

tografia, incisione e impiantazione ionica. Ciò permette di realizzare dispositivi di dimensioni ridottissime - dell'ordine dei 10 µm di lato - con una massa sensibile di meno di 1 µg, che sono utilizzabili come sensori di movimento inerziali. Naturalmente, oltre al dispositivo meccanico è possibile integrare sullo stesso chip i circuiti elettronici di controllo, d'acquisizione e di condizionamento del segnale, sia analogici sia digitali. Il sensore MEMS vero e proprio è costituito da due condensatori collegati in un half-bridge: un'accelerazione muove la massa sensibile (che costituisce una delle armature dei condensatori) facendo variare la capacità dei due condensatori. Le capacità di questi condensatori è dell'ordine del pF, mentre la massima variazione di capacità è dell'ordine di 10-100 fF. Lo sbilanciamento dell'half-bridge può essere misurato integrando la carica accumulata dai condensatori quando questi sono sottoposti a dei brevi impulsi di tensione. Gli accelerometri MEMS possiedono una serie di parametri specifici che saranno approfonditi di seguito.

Sensibilità: è la misura del guadagno del sensore sottoposto ad un'accelerazione di 1g. Può essere determinata sperimentalmente con facilità ruotando l'asse che ci interessa del MEMS a -90° (cioé verso terra) e rilevando il valore in uscita, quindi ruotandolo a 90°, vale a dire verso il cielo, e rilevando il nuovo valore in uscita. Abbiamo così applicato un range di accelerazione di 2g, da cui possiamo calcolare la sensibilità del MEMS. Questo parametro varia leggermente in funzione della temperatura.

Zero-g bias: è il livello di tensione in uscita dal sensore quando è applicata un'accelerazione nulla (0 g). Può essere determinata, per gli assi X ed Y, ponendo il sensore su di un piano orizzontale. Il valore nominale è generalmente pari a ½Vdd, vale a dire circa 1.65 V per un sensore alimentato a 3.3V. Lo spostamento dal valore nominale è denominato *zero-g offset*, ed è importante che sia il più piccolo possibile per alcuni tipi di applicazione, come vedremo. Quest'offset varia anche leggermente in funzione della temperatura.

Larghezza di banda: nel caso degli accelerometri con risposta in DC, come gli accelerometri MEMS, corrisponde alla frequenza massima dell'accelerazione da misurare. Generalmente è consigliabile diminuire la larghezza di banda del sensore al minimo possibile, in funzione della particolare applicazione, utilizzando un filtro passa-basso in uscita. È anche importante che la frequenza di taglio di questo filtro sia ben inferiore alla frequenza di risonanza del sensore, ricavabile dai dati caratteristici forniti dal produttore.

Rumore: il rumore prodotto dai sensori MEMS è un rumore bianco con distribuzione gaussiana. Questo comporta che è possibile diminuire il rumore limitando la larghezza di banda del sensore.

Risoluzione: la minima accelerazione che è possibile rilevare. Il rumore ha un impatto importante sulla risoluzione del sensore ed è quindi importante assicurarsi che un dato modello sia adatto all'applicazione che si vuole realizzare. Per fare un esempio, il valore rilevato da un sensore MEMS avente un rumore tipico di 50 g/ (Hz), se la sua banda viene limitata a 100 Hz, avrà una risoluzione di 2 mg, corrispondenti ad una variazione dell'inclinazione di 0.12°. Si noti che nel calcolo è stato considerato il rumore picco-picco, con un valore stimato pari a 4 volte il rumore RMS. Un altro fattore che determina la risoluzione è il convertitore A/D, che può essere incluso nello stesso chip, nel caso di sensori digitali, o esterno. Ad esempio, un convertitore A/D da 10 bit funzionante con una Vref = 3.3 V, nel caso di un MEMS con sensibilità di 660 mV/g, avrà una risoluzione di 4.88 mg/bit corrispondente a 0.28°/bit.

RILEVAZIONE DI VELOCITÀ E POSIZIONE CON I MEMS

Un sensore MEMS possiede uno o più assi lungo i quali può rilevare le accelerazioni, ad esempio un sensore a 3 assi possiederà una terna di assi ortogonali fra loro, come si può vedere in **figura 1**. Dato che l'accelerazione è un vettore, l'accelerazione misurata è pari al prodotto scalare della stessa nella direzione di ciascun asse di rilevazione del MEMS: ad esempio l'accelerazione A avrà una componente lungo l'asse X del sensore pari a Ax = A*cos(a) dove a è l'angolo tra A e l'asse X, come si può vedere in **figura 2**. La misurazione di velocità e posizione come già detto è eseguita integrando una volta l'accelerazione misurata per ottenere la velocità o due volte per ottenere la posizione. Se è necessario conoscere la posizione assoluta deve essere nota anche la posizione iniziale oltre alla velocità iniziale. L'equazione per il calcolo della velocità è la seguente

$$V_x = V_{x0} + \int_{t0}^{t} A_x dt$$

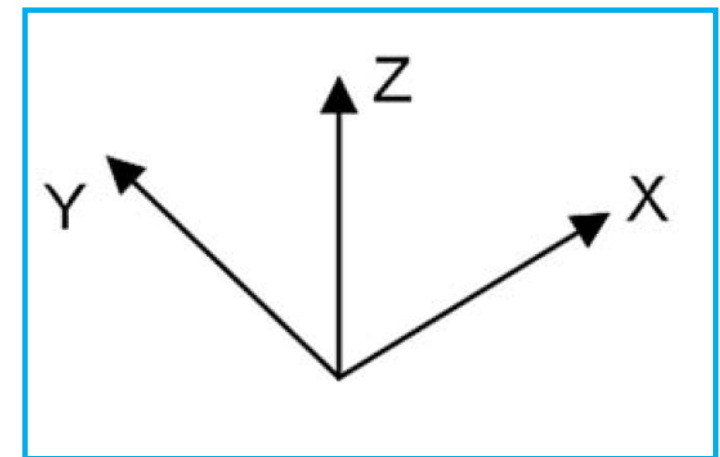
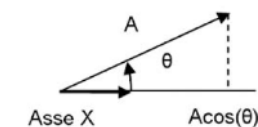



Figura 1: disposizione degli assi di un accelerometro MEMS a 3 assi.

Mentre quella per il calcolo della posizione a partire dalla velocità è

Per questo tipo di applicazione sono particolarmente importanti la precisione del dispositivo e la stabilità, in quanto ogni variazione dell'offset viene rilevata come se fosse un'accelerazione. Dato che piccole variazioni dell'accelerazione possono portare a grandi variazioni nella velocità/posizione calcolate, è anche necessario che il dispositivo abbia una risoluzione adeguata oltre che un basso rumore. I MEMS rilevano la forza di gravità come se fosse un'accelerazione, pari a 9.81 m/s2 sul li-

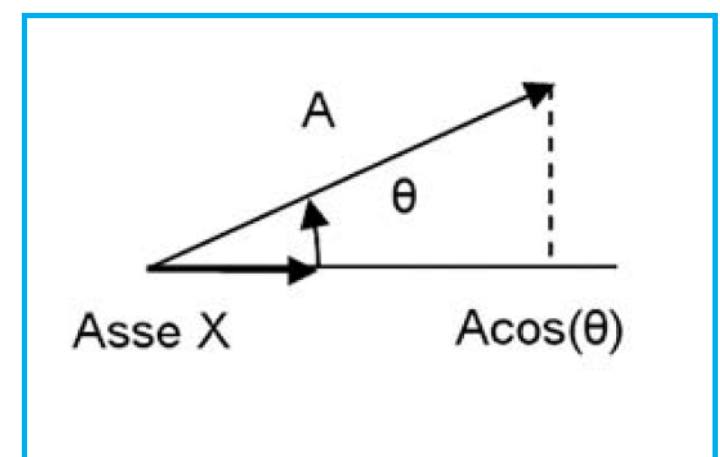



Figura 2: componente dell'accelerazione A nella direzione di un asse del sensore.

vello del mare. Quindi un'inclinazione è rilevata come se fosse un'accelerazione e in definitiva una variazione della posizione. Generalmente si utilizzano dei giroscopi, anch'essi eventualmente realizzati con tecnologia MEMS, per poter discriminare tra un'accelerazione lineare ed una rotazione. Le applicazioni di questo tipo richiedono nel caso più generale 6 gradi di libertà che corrispondono ai 3 assi di accelerazione lineare ed a 3 assi di rotazione: sono quindi necessari 3 accelerometri (o un accelerometro a 3 assi) per rilevare le accelerazioni e 3 giroscopi per rilevare le rotazioni.

RILEVAZIONE DI SHOCK MECCANICIE DI VIBRAZIONI

Particolari applicazioni, come quelle nelle quali è necessario discriminare tra semplici accelerazioni 'normali' e accelerazioni da impatto, richiedono un'elaborazione sofisticata dei segnali rilevati. Ad esempio è possibile utilizzare un'analisi spettrale dell'accelerazione per riconoscere un determinato pattern. Se la larghezza di banda è limitata (generalmente siamo nell'ordine di poche decine-centinaia di Hz) è possibile implementare una FFT anche su un micro ad 8 bit, purché dotato almeno di un moltiplicatore hardware. Altre applicazioni possibili sono quelle riguardanti la rilevazione di vibrazioni per determinare lo stato di salute di macchinari elettromeccanici come ventole, pompe, ecc... In quest'ambito è necessario tenere conto in particolare della larghezza di banda del sistema e del rumore, il cui impatto come abbiamo visto è inversamente proporzionale alla larghezza di banda.

RILEVAZIONE DELL'INCLINAZIONE (TILT SENSING)

Una delle applicazioni più frequenti degli accelerometri riguarda la rilevazione dell'inclinazione o dell'orientamento. Grazie al fatto che i MEMS sono sensibili alla forza di gravità, è possibile rilevare l'inclinazione intorno ad uno degli assi. In questo caso ci troviamo di fronte al problema opposto rispetto a quello che avevamo evidenziato in precedenza nel caso della misura di velocità: un'accelerazione viene rilevata come se fosse un'inclinazione. Anche in questo caso per poter discriminare tra accelerazione ed inclinazione è possibile utilizzare un giroscopio o più accelerometri disposti correttamente. Ad esempio se disponiamo di un accelerometro a 3 assi possiamo usare a questo scopo l'asse Z. Gli accelerometri possono rilevare il beccheggio (pitch, rotazione attorno all'asse X) ed il rollio (roll, rotazione attorno all'asse Y) ma non l'imbardata (yaw, rotazione attorno all'asse Z) in quanto in questo caso la forza di gravità è ininfluente. In quest'ultimo caso è necessario quindi un giroscopio o più accelerometri con un software adatto. L'inclinazione di uno degli assi dell'accelerometro rispetto all'orizzonte viene rilevata come un'accelerazione pari a

$$A_i = \left(\frac{V_{out} - V_{zero-g}}{Sensitivity} \right)$$

da cui l'inclinazione:

$$\theta = \arcsin\left(\frac{V_{out} - V_{zero-g}}{Sensitivity} \right)$$

valida per un angolo compreso tra -90° e +90°. In **tabella 1** sono riportati i valori in uscita dall'asse X di un MEMS per alcuni valori notevoli dell'inclinazione: notiamo che la sensibilità del MEMS non è un valore costante ma raggiunge un massimo intorno agli 0° d'inclinazione ed un minimo a -90° e +90°. In pratica ciò lo renderebbe utilizzabile solo per rotazioni comprese tra +-45°, ma combinando i valori rilevati da due assi, come gli assi X e Z, è possibile ottenere una sensibilità costante pari al valore massimo di 17.45 mg/°. Si osservi che i valori così calcolati sono indipendenti dal particolare modello di sensore poiché discendono da calcoli trigonometrici sempre validi. Nella misura dell'inclinazione, sono importanti diversi parametri caratteristici:

- precisione, ripetibilità e bassa variazione dello zero-g bias;
- risoluzione, determinata a sua volta dal rumore.

Tabella 1

ORIENTAMENTO ASSE X (rispetto all'orizzonte) (°)	ACCELERAZIONE RILEVATA DALL'ASSE X (g)	SENSIBILITÀ PER GRADO D'INCLINAZIONE (mg/°)
-90	-1	-0.15
-75	-0.966	4.4
-60	-0.866	8.6
-45	-0.707	12.2
-30	-0.500	15.0
-15	-0.259	16.8
0	0	17.45
15	0.259	16.9
30	0.500	15.2
45	0.707	12.4
60	0.866	8.9
75	0.966	4.7
90	1	0.15

UN ACCELEROMETRO MEMS: LIS3L02AL

L'accelerometro MEMS che useremo nel nostro progetto, è prodotto da ST Microelectronics . Si tratta di un accelerometro a 3 assi con uscita analogica capace di rilevare accelerazioni comprese tra ±2g. In **figura 3** possiamo vedere lo schema a blocchi del componente, mentre in **tabella 2** sono riportate le caratteristiche principali di questo componente. Il dispositivo comprende al suo interno i circuiti di pilotaggio dei sensori, l'amplificatore di carica, 3 S&H, un sistema di Self-Test ed altri circuiti di controllo. L'amplificatore di carica opera a 200 KHz, mentre i Sample & Hold campionano la tensione in uscita dall'amplificatore alla frequenza di 66 KHz: l'oversampling serve principalmente a ridurre il rumore. Lo zero-g bias e la sensibilità so-

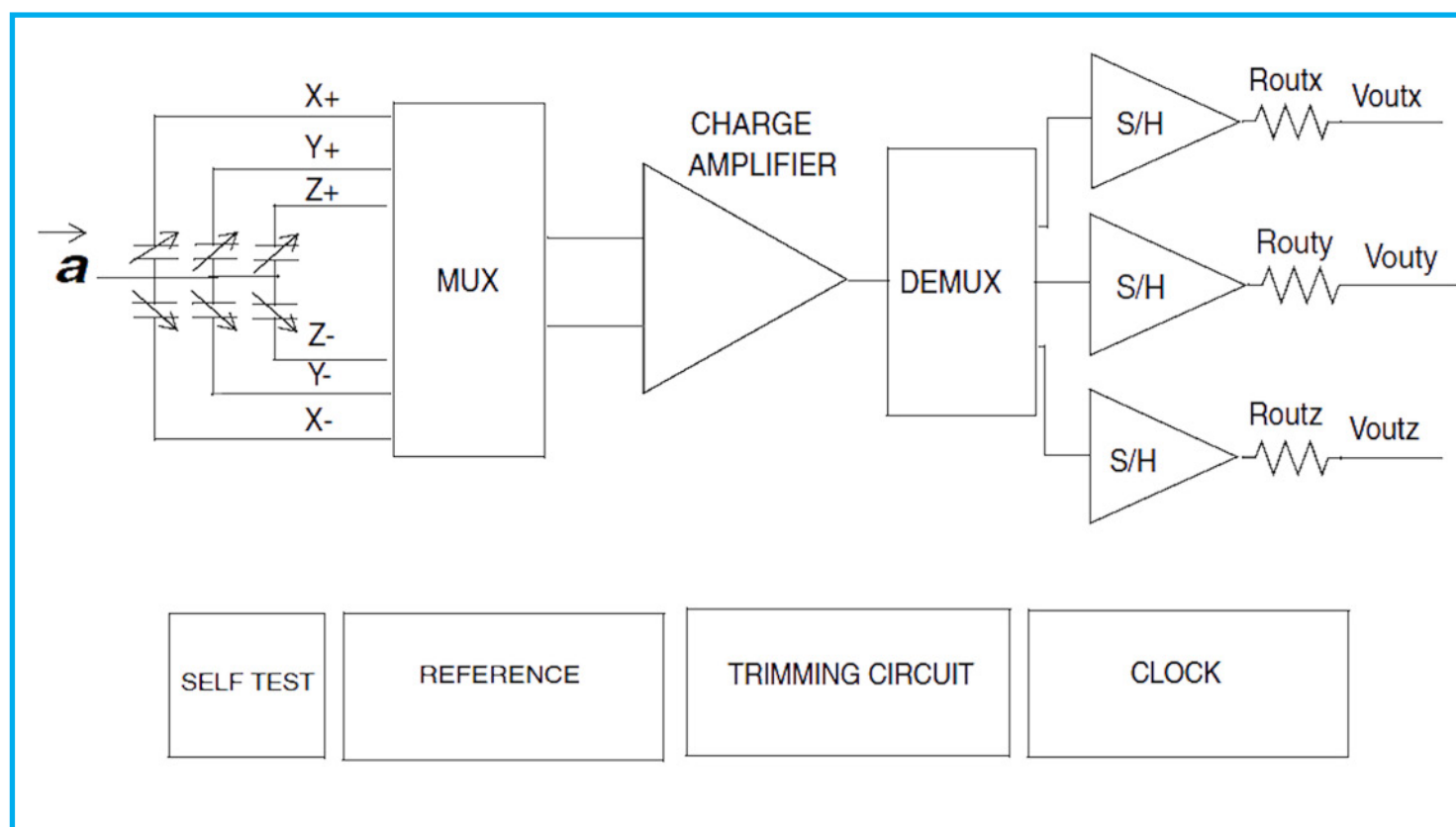



Figura 3: schema a blocchi dell'accelerometro a 3 assi LIS3L02AL

no proporzionali alla tensione di alimentazione, quindi questi parametri variano linearmente con le variazioni di questa tensione. Questi stessi parametri sono calibrati in fabbrica ed i valori corrispondenti memorizzati in una struttura non volatile che viene letta all'accensione del dispositivo, per cui non è necessario ricorrere alla sua taratura. Il MEMS LIS3L02AL possiede un circuito di mediante il quale è possibile rilevare il corretto funzionamento sia della parte meccanica che elettronica. Questo sistema è costituito essenzialmente da altri due condensatori per ogni asse che fungono questa volta da attuatori: ponendo il pin *ST* a livello alto, questi condensatori sono caricati generando una forza elettrostatica che induce un'accelerazione aggiuntiva sulla massa sensibile del sensore.
L'uscita del sensore sarà quindi data dalla somma algebrica di questa accelerazione con quella esterna, permettendo così di testarlo. I 3 S & H sono collegati alle uscite degli assi X, Y e Z tramite una resistenza che costituisce parte di un filtro passa-basso.
È necessario completare il filtro aggiungendo un condensatore esterno di valore opportuno tra l'uscita e la massa. La frequenza di taglio nominale del filtro sarà data dalla formula seguente:

$$f_t = \frac{1.45 * 10^{-6}}{C_{load}}$$

Il valore minimo di Cload è di 1 nF, corrispondente ad una ft = 1.5 KHz, che come risulta dai dati caratteristici corrisponde alla frequenza di risonanza del sensore. In **figura 4** sono riportate la disposizione degli assi e le connessioni del dispositivo.

Tabella 2

PARAMETRO	VALORE TIPICO
(Vdd = 3.3 V, Ta = 25°C se non diversamente specificato)	
Tensione di alimentazione, Vdd	2.4 ÷ 3.6 V
Range di accelerazione	±2g
Sensitivity	(Vdd/5) V/g
Sensitivity @ 3.3 Vdd	660 mV/g
Zero-g bias	(Vdd/2) V
Non linearità Assi X e Y	±0.3%
Asse Z	±0.5%
Densità di rumore del sensore	50 g/ Hz
Frequenza di risonanza del sensore	1.5 KHz
Impedenza d'uscita, Rout	110 Kohm

AIR MOUSE

Per realizzare l'Air Mouse verrà usato un accelerometro MEMS del tipo LIS3L02AL per emulare il comportamento del puntatore di un mouse. Le accelerazioni rilevate dagli assi X ed Y del MEMS saranno acquisite tramite un microcontrollore dotato di interfaccia USB 2.0 Full Speed (12 mb/s). Tramite questa interfaccia il sistema potrà essere collegato al PC che lo rileverà come un dispositivo USB di classe HID (*Human Interface Device*).

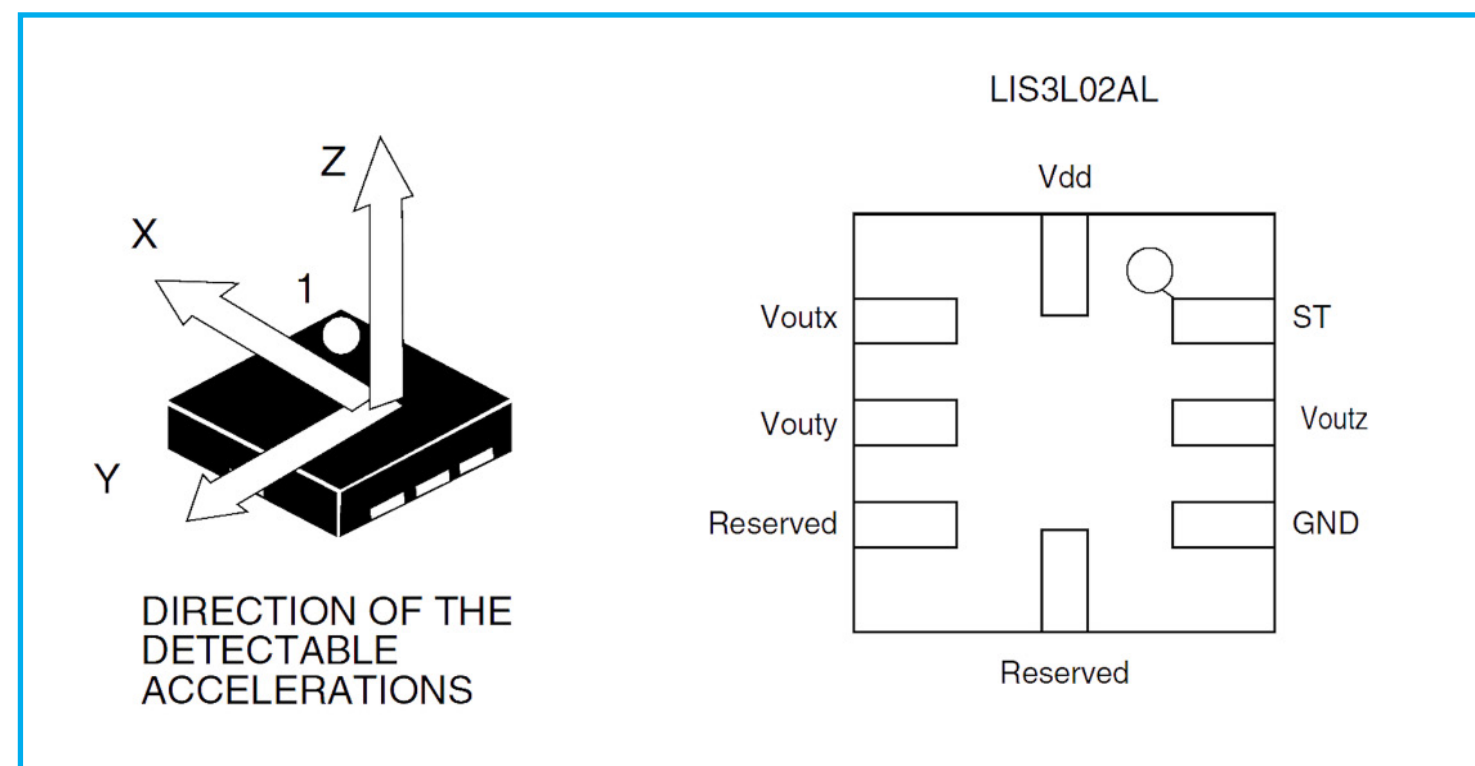



Figura 4: disposizione degli assi e connessioni del MEMS LIS3L02AL (visto dal basso).

L'hardware del sistema

Con riferimento allo schema elettrico in **figura 5**, le uscite degli assi X ed Y del sensore MEMS sono collegati a due ingressi del microcontrollore PIC18LF2550 che li convertirà tramite il convertitore A/D a 10 bit integrato.

I condensatori C1 e C2 completano i filtri passa-basso delle due uscite del MEMS: il valore prescelto di 100 nF limita la banda passante a circa 15 Hz, valore adeguato per la nostra applicazione. Il Clock del PIC è generato tramite il quarzo XTAL1 da 4 Mhz ed il PLL interno che genera le frequenze di clock di 32 Mhz per la CPU e di 48 Mhz per l'interfaccia USB Full Speed. Si faccia attenzione alla disposizione dei terminali del connettore USB che dovrà essere di tipo A (come quello dei Flash Drive, per intenderci).

Contrariamente a quanto siamo soliti fare con i dispositivi elettronici, la corretta disposizione meccanica del MEMS è cruciale: nella nostra applicazione sarà infatti necessario porre il sensore in orizzontale e con il puntino indicatore del pin 1 rivolto in alto a sinistra, pena ottenere un comportamento che non risponde a quello voluto.

Si noti che nello schema elettrico le connessioni del MEMS sono viste dall'alto. Tutto il sistema è alimentato a 3.3 V, eventualmente ricavabili dall'interfaccia USB tramite un regolatore LVD opportuno. In questo caso sarà però necessario modificare il firmware secondo le indicazioni che potete trovare nel file di configurazione *HardwareProfile.h*.

Figura 5: schema elettrico dell'Air Mouse.

Il firmware

Potete scaricare i file dell'applicazione dal sito della rivista. Il file che ci interessa maggiormente è *MEMS.c*, un estratto del quale è visibile nel **listato 1** La funzione *MEMSInit()* contenuta in questo file esegue alcune inizializzazioni del PIC e delle strutture dati usate dall'emulatore.

La funzione *MEMSADConv()* legge il valore rilevato dall'asse X o Y del MEMS ed acquisito tramite il modulo A/D che lo converte in un valore numerico su 10 bit. La funzione *MEMSEmulateMouse()* utilizza i valori delle accelerazioni così rilevati per emulare il funzionamento del mouse: i dati elaborati saranno inviati all'host tramite l'endpoint IN utilizzando la funzione *HIDTxPacket()*dello stack USB. Il firmware fa uso dello stack USB di Microchip, scaricabile da per realizzare un dispositivo di classe HID che viene visto come un mouse standard dal sistema operativo. Una volta collegato il connettore USB ad una porta libera del PC, il dispositivo sarà riconosciuto automaticamente dal SO senza necessità di installare driver.

Muovendo il MEMS con una leggera accelerazione verso destra/sinistra il puntatore del mouse si sposterà verso destra/sinistra, mentre muovendolo avanti o indietro il puntatore si sposterà rispettivamente in basso o in alto.

Notiamo che questo comportamento è più simile a quello di un joystick che di un mouse, ma può essere facilmente modificato, così come è possibile tarare i parametri di accelerazione DELTAPOS, DELTAPOSH, DELTANEG e DELTANEGH a seconda delle esigenze.

LISTATO 1

```
void MEMSEmulateMouse(void)
{
static int MEMSValueXOld=512, MEMSValueYOld=512 ;
// Legge il valore misurato dal MEMS
// compreso tra 0 e 1023
MEMSValueX = MEMSADConv(ADC_CH0);
MEMSValueY = MEMSADConv(ADC_CH1);
buffer[0] = 0; // Pulsanti del mouse
// X-Vector
// si sposta a sinistra
if (((MEMSValueX< DELTANEG) & (MEMSValueXOld<=DELTAPOS)) |
((MEMSValueX< DELTANEGH) & (MEMSValueXOld>DELTAPOS))) {
MEMSValueXOld = MEMSValueX;
```

(segue)

```
buffer[1] = -1;
}
// si sposta a destra
else if (((MEMSValueX> DELTAPOS) & MEMSValueXOld>=DELTANEG))
|((MEMSValueX> DELTAPOSH) & (MEMSValueXOld<DELTANEG))) {
MEMSValueXOld = MEMSValueX;
buffer[1] = 1;
}
else
buffer[1] = 0;
// Y-Vector
// si sposta in basso
if (((MEMSValueY< DELTANEG) & (MEMSValueYOld<=DELTAPOS)) |
((MEMSValueY< DELTANEGH) & (MEMSValueYOld>DELTAPOS))) {
MEMSValueYOld = MEMSValueY;
buffer[2] = +1;
}
// si sposta in alto
else if (((MEMSValueY> DELTAPOS) &(MEMSValueYOld>=DELTANEG))
|((MEMSValueY> DELTAPOSH) & (MEMSValueYOld<DELTANEG))) {
MEMSValueYOld = MEMSValueY;
buffer[2] = -1;
}
else
buffer[2] = 0;
// MEMS e' fermo
// invia i parametri tramite l'endpoint IN
if(HIDTxHandleBusy(lastTransmission) == 0)
{
hid_report_in[0] = buffer[0];
hid_report_in[1] = buffer[1];
hid_report_in[2] = buffer[2];
//Invia il packet (3 byte) tramite USB.
lastTransmission = HIDTxPacket(HID_EP,
(BYTE*)hid_report_in, 0x03);
}
}
```

CIRCUITI A LED

Una panoramica sul funzionamento e la polarizzazione dei diodi LED e ben 20 circuiti applicativi

I LED è l'equivalente moderno della lampadina. Ha cambiato radicalmente, da indicatore non troppo brillante a troppo luminoso per essere osservato. Tuttavia è completamente diverso da una "lampadina". Una lampada e un dispositivo elettrico con un filamento incandescente, mentre un LED è un dispositivo elettronico. Un LED e più efficiente, produce meno calore e deve essere "pilotato" correttamente affinché non si danneggi. Questo eBook vi illustra come collegare i LED in un circuito, oltre a svariati progetti che li impiegano. Usare un LED è semplice - quando sapete come fare.

COLLEGAMENTO DI UN LED

In un circuito un LED deve essere collegato correttamente e avere una resistenza di limitazione della corrente. Con riferimento alla figura 1, il LED del primo schema non si illumina perché un LED richiede circa 1,7V e la batteria fornisce soltanto 1,5V. Il LED nel secondo schema viene danneggiato perché richiede 1,7V e le due batterie forniscono 3V. Per limitare la corrente a circa 25mA e la tensione a 1,7V è necessaria una resistenza, come illustrato sul terzo schema. Il quarto schema è il circuito del layout #3 che illustra il simbolo del LED, della resistenza, della batteria e di come i tre sono connessi. Il LED del quinto schema non funziona perché è collegato al contrario.

CADUTA DI TENSIONE TIPICA

Quando un LED viene collegato correttamente all'interno di un circuito si stabilisce

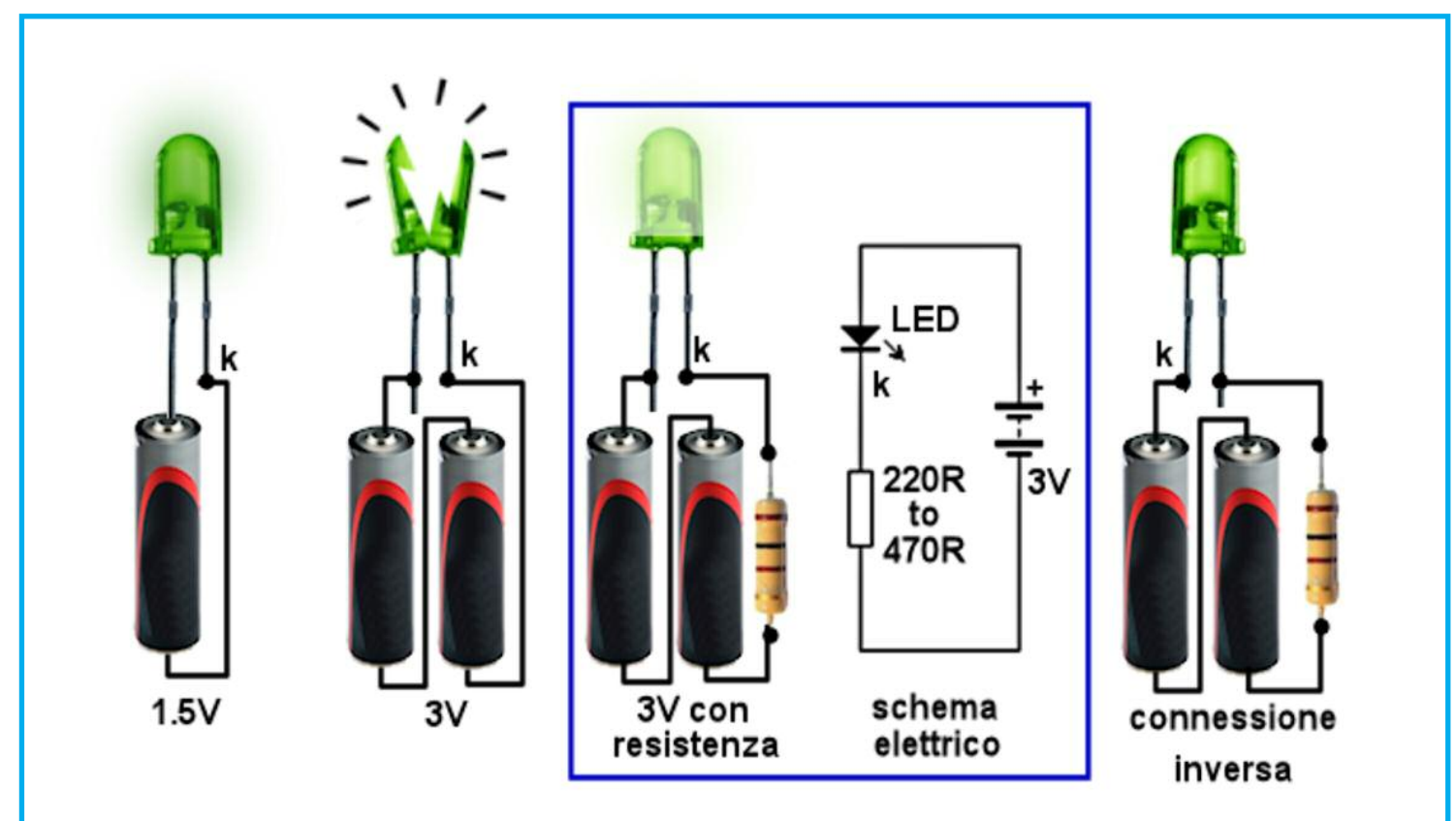



Figura 1: collegare un LED alla batteria

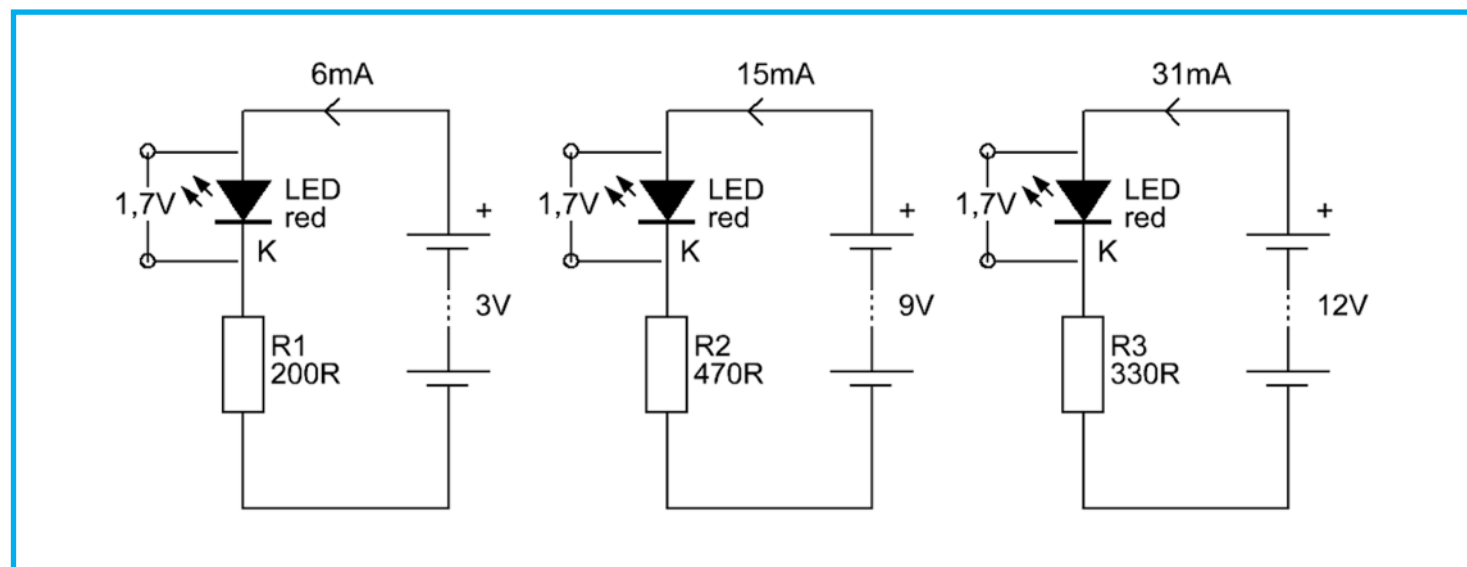



Figura 2: schemi di connessione di un LED

una tensione ai suoi reofori, chiamata CADUTA DI TENSIONE TIPICA. Un LED deve essere alimentato con una tensione che sia più alta della sua " TENSIONE TIPICA" attraverso una resistenza (definita RESISTENZA DI CADUTA o RESISTENZA DI LIMITAZIONE DI CORRENTE) in modo che il LED funzioni in modo corretto e possa fornire almeno da 10.000 a 50.000 ore di illuminazione.
Un LED funziona in questo modo: LED e resistenza vengono collegati in serie e alimentati da una tensione (figura 2). Quando questa inizia a crescere rispetto a 0V, non accade nulla finché il suo valore raggiunge all'incirca 1,7V. A questa tensione un LED rosso comincia appena ad illuminarsi. Con l'aumentare della tensione la differenza di potenziale sul LED rimane a 1,7V, ma la corrente aumenta e il LED diventa più luminoso.
Ora rivolgiamo la nostra attenzione alla corrente che scorre nel LED. Aumentando a 5mA, 10mA, 15mA, 20mA la luminosità aumenterà e a 25mA raggiungerà il massimo. Aumentando ulteriormente la tensione di alimentazione, il colore del LED varierà leggermente, ma il cristallo al suo interno inizierà a surriscaldarsi riducendone considerevolmente la vita.
Questo era solo un semplice esempio di come ciascun LED abbia una propria TENSIONE CARATTERISTICA DI CADUTA e una corrente massima diversa.
Sullo schema sottostante vediamo un LED con alimentazione a 3V, 9V e 12V. Le resistenze di limitazione sono diverse e il primo circuito assorbe 6mA, il secondo 15mA e il terzo 31mA. Ma la tensione ai capi del LED rosso è la stessa in tutti i casi. Questo avviene perché il LED crea la TENSIONE CARATTERISTICA DI CADUTA e questa non varia. Non importa se la resistenza è collegata sopra o sotto il LED. I circuiti funzionano allo STESSO modo.

TENSIONE RESIDUA

Ora occupiamoci della resistenza. All'aumentare della tensione di alimentazione, la tensione ai capi del LED rimarrà costantemente a 1,7V (per un LED rosso) e la tensione in eccesso andrà a cadere ai capi

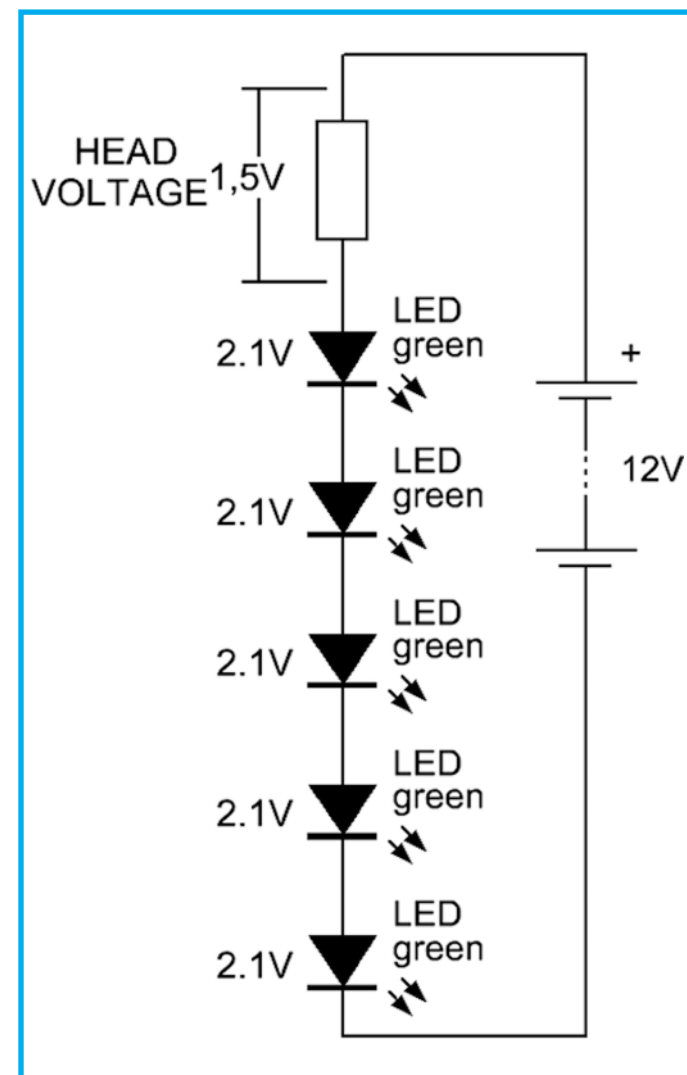



Figura 3: gestione della tensione residua

della resistenza. L'alimentazione può assumere qualsiasi valore da 2V a 12V o più. In questo caso, la resistenza farà cadere da 0,3V a 10,3V. Questa si definisce TENSIONE RESIDUA - o "HEAD-VOLTAGE". Lo schema di figura 3 mostra la TENSIONE RESIDUA: La tensione che cade ai capi di questa resistenza, combinata con la corrente, costituisce energia dispersa e dovrebbe essere tenuta al minimo, ma una TENSIONE RESIDUA troppo ridotta (come 0,5V) non è consigliabile. La tensione residua dovrebbe essere almeno 1,5V - e questo si verifica soltanto se l'alimentazione è costante. L'head voltage dipende dalla tensione di alimentazione. Se l'alimentazione è regolata e non subisce diminuzioni o aumenti, la tensione residua può essere ridotta (1,5V minimo).
Ma la maggior parte delle alimentazioni deriva da batterie e la tensione scende durante il consumo. Qui c'è un esempio del problema:

Alimentazione: 12V
7 LED rossi in serie = 11,9V
Caduta su resistenza = 0,1V
Non appena l'alimentazione scende a 11,8V, nessun LED emetterà luce.

Altro esempio:

Alimentazione 12V
5 LED verdi in serie @ 2,1V = 10,5V
Caduta su resistenza = 1,5V
La tensione di batteria può scendere a 10,5V

Ma guardiamo la situazione più da vicino.
Supponiamo una corrente@ 12V=25mA.
Come la tensione diminuisce, la corrente farà altrettanto.
A 11,5V, la corrente sarà 17mA
A 11V, la corrente sarà 9mA
A 10,5V, la corrente sarà zero
Potete notare che la caduta di tensione sostenibile è solo di circa 1V.
Molte batterie scendono di 1V e mantengono oltre l' 80% della loro energia rimanente. Per questo motivo dovete progettare il vostro circuito in modo che abbia un'ampia HEAD VOLTAGE.

PROVA DI UN LED

Se non si riesce a identificare il catodo di un LED, collegate 3 pile in serie a una resi-

stenza da 220Ohm e illuminatelo. 4,5V permettono di provare tutti i tipi di LED dato che i LED bianchi richiedono fino a 3,6V. Non usate un multimetro, dato che alcuni di essi hanno solo due batterie e questo non permetterà di accendere tutti i tipi di LED. Inoltre, il puntale negativo del multimetro è collegato al positivo delle batterie (all'interno dello strumento) per la misura di resistenze- facendovi ottenere una identificazione scorretta del catodo.

IDENTIFICAZIONE DI UN LED

Un LED non ha terminali "Positivi" o "Negativi". Ha un reoforo identificato come "Catodo" o "Kathode" o "k". Questo viene identificato con una smussatura a lato del LED e/o con il terminale più corto. Questo capo è collegato al lato a 0V del circuito o vicino ad esso (se il LED è collegato ad altri componenti).
Molti LED hanno un lato piatto che identifica il catodo. Alcuni LED a montaggio superficiale hanno un punto o una marcatura per identificare il catodo e alcuni hanno un intaglio su un lato.

PILOTGGIO IN CORRENTE

Il LED è definito come DISPOSITIVO PILOTATO IN CORRENTE. Ciò significa che la sua luminosità e determinata dall'intensità della corrente che lo percorre. La luminosità di un LED può essere alterata dall'aumento o dalla diminuzione della corrente. L'effetto non è lineare ed è consigliabile eseguire degli esperimenti per determinare la corrente migliore per il livello di illuminazione richiesto. I LED tradizionali e super-luminosi si illumineranno a 1mA o meno, per cui la qualità di un LED ha molto a che vedere con la luminosità. L'utilizzo di molti LED avviene a 17mA. Questo sembra essere il valore migliore per la maggior parte di essi. LED da 1mA a 5mA Alcuni LED emetteranno luce a 1mA. Questi sono i LED ad "alta qualità" o "alta luminosità" e l'unico modo di verificare questa caratteristica è di provarli @1mA come illustrato sotto. I LED a 5V Alcuni fornitori e qualche sito web citano un LED bianco o blu da 5V. Alcuni LED hanno una piccola resistenza interna e possono essere collegati all'alimentazione di 5V. Sono molto rari. Alcuni siti web suggeriscono il collegamento di un LED bianco alla tensione di 5V. Questi LED hanno una caduta di tensione caratteristica di 3,6V e non dovrebbero essere collegati direttamente a tensioni superiori a questo valore. L'unico LED con una resistenza interna è il LED LAMPEGGIANTE. Questi LED possono essere alimentati con tensioni da 5V a 12V e lampeggiano approssimativamente a 2Hz. MAI presupporre che un LED abbia una resistenza interna. Aggiungete sempre una resistenza in serie. Alcuni LED ad alta luminosità sono progettati per lavorare a 12V. Questi LED hanno un completo circuito interno che regola correttamente la corrente del LED. Questo tipo di dispositivo non è descritto in questo ebook.

LED IN SERIE

I LED possono essere collegati in serie a condizione che si tenga conto di alcune caratteristiche. Il componente principale da includere è una resistenza di limitazione-corrente. Un LED e una resistenza sono chiamati stringa. Una stringa può avere 1, 2, 3 o più LED. I punti da osservare sono tre:

1. MASSIMA CORRENTE attraverso ciascuna stringa = 25mA.
2. La CADUTA DI TENSIONE CARATTERISTICA deve essere conosciuta, di modo che in ogni stringa si possa usare il numero corretto di LED.
3. Una RESISTENZA DI CADUTA deve essere inserita in ogni stringa.

LED IN PARALLELO

I LED "generano" o "possiedono" o "creano" una tensione ai loro capi chiamata CADUTA DI TENSIONE CARATTERISTICA (quando sono inseriti correttamente in un circuito). Questa tensione è generata dal tipo di cristallo e differisce per ogni colore, così come per la "qualità" del LED (come alta luminosità, ultra-alta luminosità etc.). Questa caratteristica non può essere variata MA presenta piccolissime variazioni da un LED a un altro dello stesso lotto. E aumenta leggermente all'aumentare della corrente. Per esempio, ci saranno delle differenze di qualcosa come 0,2V per i LED rossi e 0,4V per i bianchi dello stesso lotto e aumenteranno di 0,5v quando la corrente è aumentata dal minimo al massimo. Potete verificare 100 LED bianchi @15mA e misurare la CADUTA DI TENSIONE CARATTERISTICA per verificare questo intervallo. Se prendete 2 LED con identica CADUTA DI TENSIONE CARATTERISTICA e li collegate in parallelo, assorbiranno entrambi la medesima corrente. Ciò significa che i 30mA attraverso la resistenza di limitazione saranno divisi in 15mA per ciascun LED. Tuttavia, se un LED possiede una più alta CADUTA DI TENSIONE CARATTERISTICA, im-

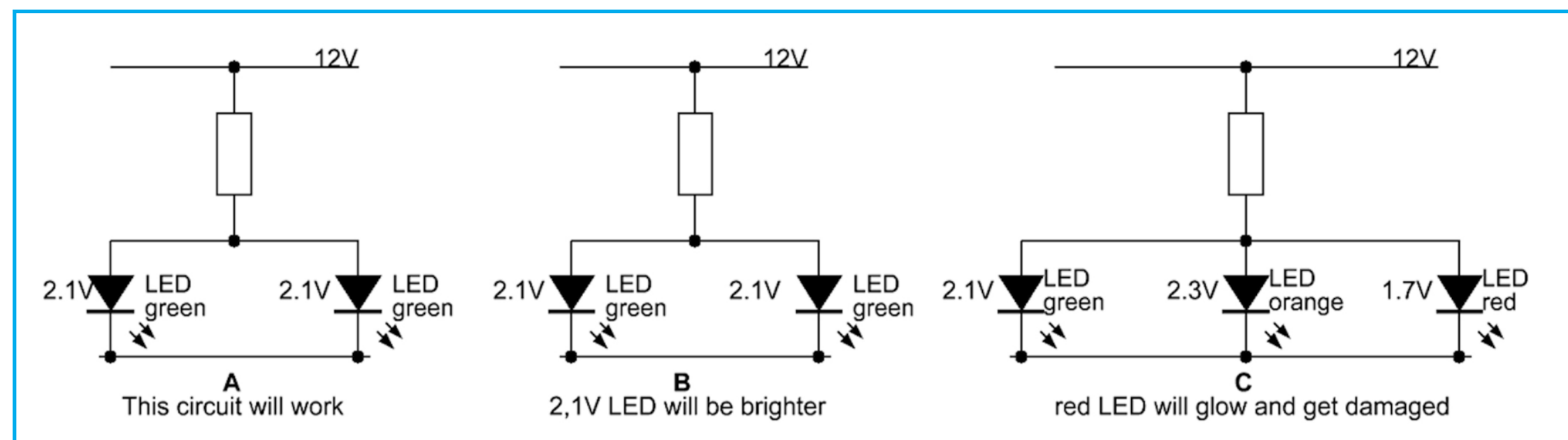



Figura 4: collegamento in parallelo di LED

porrà una corrente minore e l'altro LED assorbirà considerevolmente di più. Comunque non avete modo di stabilire la "ripartizione-delle-correnti" in una stringa di LED in parallelo. Se collegate 3 o più LED in parallelo, un LED inizierà ad assorbire più corrente, si surriscalderà e vi troverete molto rapidamente un LED guasto. Non appena un LED si guasta, gli altri assorbiranno una corrente maggiore e l'insieme del gruppo dei LED inizierà l'auto-distruzione. In ogni caso il collegamento dei LED in PARALLELO dovrebbe essere evitato. Lo schema A di figura 4 mostra due LED verdi in parallelo. Il collegamento funzionerà a condizione che la caduta di tensione su ciascun LED sia la medesima. Nello schema B la caduta di tensione caratteristica è leggermente diversa per il secondo LED e il primo LED verde si illuminerà più intensamente. Nello schema C i tre LED hanno cadute di tensione diverse e il LED rosso si illuminerà molto intensamente, mentre gli altri due LED non emetteranno luce. Tutta la corrente scorrerà attraverso il LED rosso che ne risulterà danneggiato. Il motivo per cui il LED rosso si illuminerà intensamente è questo: possiede la più bassa caduta di tensione caratteristica e stabilirà 1,7V per il gruppo dei tre LED. I LED verde e arancio a questa tensione non si accenderanno e così tutta la corrente dalla resistenza di limitazione scorrerà verso il LED rosso, distruggendolo.

LA RESISTENZA

Il valore della resistenza di limitazione può essere ricavato dalla Legge di Ohm. Questi sono i 3 passi:

1. Sommare le tensioni di tutti i LED della stringa. es.: 2,1V + 2,3V + 2,3V + 1,7V = 8,4V
2. Sottrarre la tensione dei LED da quella di alimentazione. es.: 12V – 8,4V = 3,6V
3. Dividere 3,6V (o la tensione che vi risulta) per la corrente della stringa.
   per 25mA: 3,6/0,025 = 144 ohm
   per 20mA: 3,6/0,02 = 180 ohm
   per 15mA: 3,6/0,015 = 250 ohm
   per 10mA: 3,6/0,01 = 360 ohm

Questo è il valore della resistenza di limitazione.

SALDARE UN LED

I LED sono i componenti più sensibili al calore in assoluto. Quando saldate un LED a montaggio superficiale, dovreste trattenerlo con delle pinzette e "puntare" un lato. Quindi aspettate che il LED si raffreddi e saldate rapidamente l'altro lato. Aspettate ancora pochi secondi e saldate completamente il primo lato. Verificate l'accensione di ciascun LED con 3 pile in serie e una resistenza 220R. Se avete surriscaldato il LED, la sua luminosità sarà ridotta, o il colore un po' diverso, o potrebbe non funzionare affatto. Sono estremamente sensibili al calore, fondamentalmente perché il cristallo è molto vicino alla punta del saldatore.

LED ULTRA-LUMINOSI

I LED sono diventati più efficienti negli ultimi 25 anni. Alle origini un LED rosso e metteva 17mcd @20mA. Questi LED oggi emettono da 1.000mcd a 20.000mcd @20mA. Ciò significa che potete ridurre la corrente e produrre ancora luce. Alcuni LED funzionano con una corrente ridotta anche a 1mA.

LED COME RILEVATORI DI LUCE

I LED possono anche essere utilizzati per rilevare la luce. I LED verdi sono i migliori, comunque tutti i LED potranno rilevare la luce e produrre una tensione equivalente alla loro TENSIONE CARATTERISTICA DI CADUTA, a condizione che ricevano luce a sufficienza. La corrente che producono è piuttosto limitata, anche se i LED ad alta e altissima luminosità ne producono di più, grazie al fatto che il loro cristallo è più efficiente nel processo di conversione.

LED LED LED

Vi sono centinaia di circuiti che impiegano un LED o pilotano un LED o fanno lampeggiare un LED e in ogni caso tutti circuiti di questo eBook sono diversi. Alcuni fanno lampeggiare un LED su un'alimentazione a 1,5V, altri utilizzano correnti molto deboli, altri fanno lampeggiare molto intensamente il LED e altri ancora utilizzano un LED lampeggiante per stabilire la frequenza. Da ognuno di questi circuiti imparerete qualcosa. Molti sono interessanti e tanti sorprendenti. Alcuni di questi si possono aggiungere ad altri circuiti per crearne altri, più complessi.

ALIMENTARE UN PROGETTO

Il modo più sicuro per alimentare un progetto è con una batteria. I circuiti richiedono una tensione da 3V a 12V. Questa può essere fornita da un gruppo di pile AA in un supporto, o da una batteria a 9V per alcuni progetti. Se desiderate alimentare un circuito per un lungo periodo di tempo, vi servirà un "alimentatore." Il tipo più sicuro è il cosiddetto "a parete" (anche definito in molti altri modi..). Alcuni tra questi hanno la tensione di uscita selezionabile: 3V, 6V, 7,5V, 9V, 12V) DC con la corrente massima di 500mA. Il filo nero e il negativo e l'altro con una riga bianca (o grigio con riga nera) è il positivo. Questo è il modo più sicuro per alimentare un progetto, dato che l'isolamento dalla rete è fornito dall'adattatore stesso e non c'è possibilità di ricevere scariche elettriche. L'indicazione "500mA" rappresenta il massimo che il dispositivo può erogare: se il vostro circuito richiede appena 50mA, questa è la corrente che sarà fornita. Alcuni adattatori sono dichiarati per 300mA o 1A e altri hanno una tensione di uscita fissa. Tutti questi modelli andranno bene. Alcuni alimentatori indicano "12V AC": questi tipi non sono adatti, in quanto non possiedono la sezione con diodi e condensatore relativa alla conversione da AC a DC. Tutti circuiti di questo articolo richiedono un'alimentazione in corrente continua.

LED LAMPEGGIANTE

I 7 circuiti della figura 5 fanno lampeggiare un LED con alimentazione da 1,5V a 12V. Hanno valori differenti per efficienza ed assorbimento di corrente. Ne troverete almeno uno che soddisfi i vostri requisiti. Il modo più semplice per far lampeggiare un LED è di acquistare un LED LAMPEGGIANTE come illustrato in Figura A. Può funzionare da 3V a 9V ma non è molto luminoso, principalmente perché il LED non è ad alta efficienza. Un LED LAMPEGGIANTE si può utilizzare per far accendere un LED rosso super-luminoso, come si vede in Figura B. La Figura C mostra un LED lampeggiante che pilota un transistor amplificatore per far lampeggiare un LED bianco. Questo circuito richiede 4,5V – 6V.

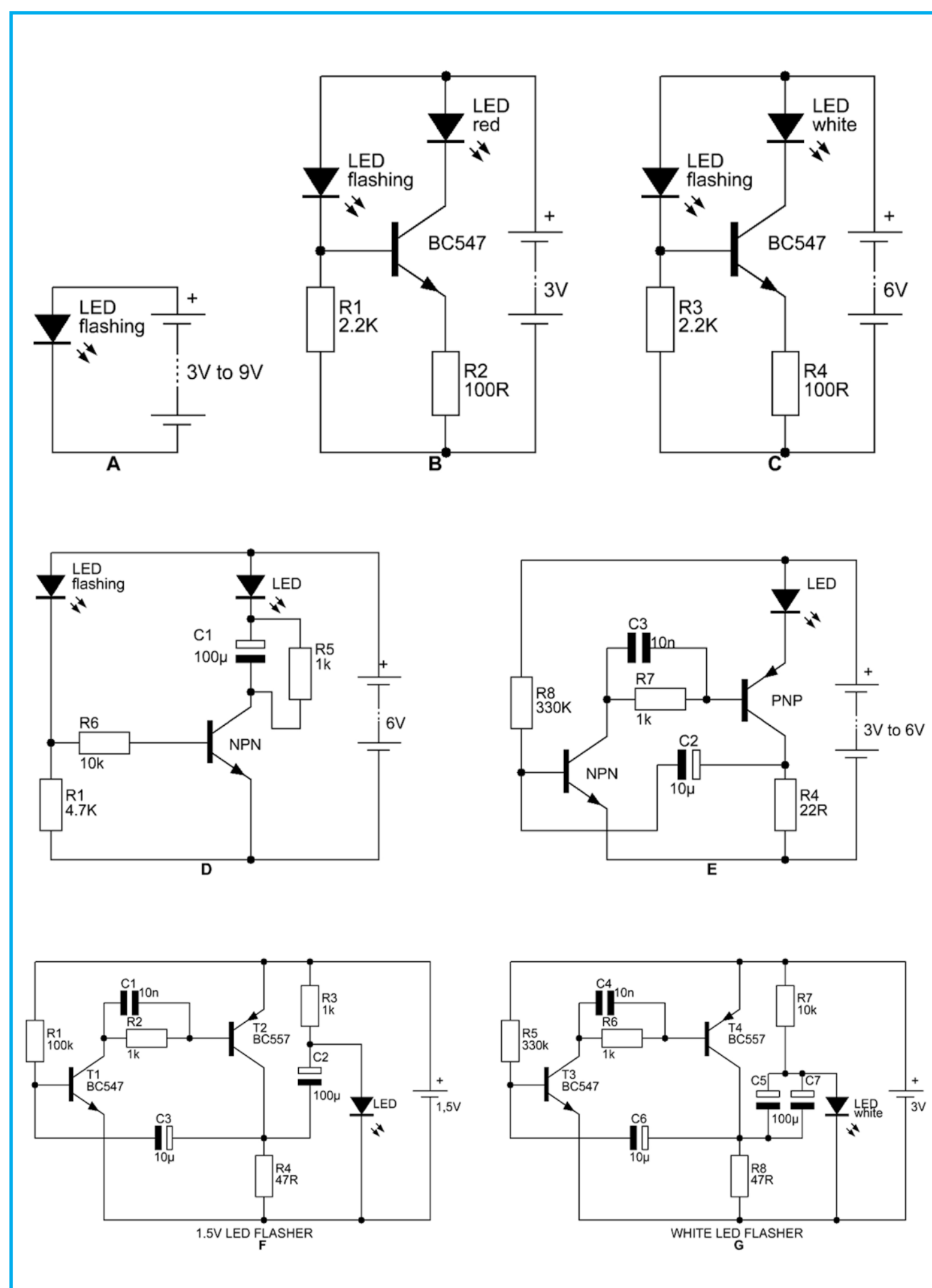



Figura 5: schemi possibili per un LED lampeggiante

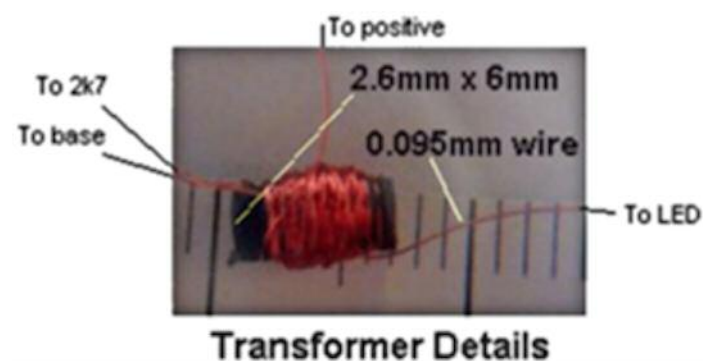
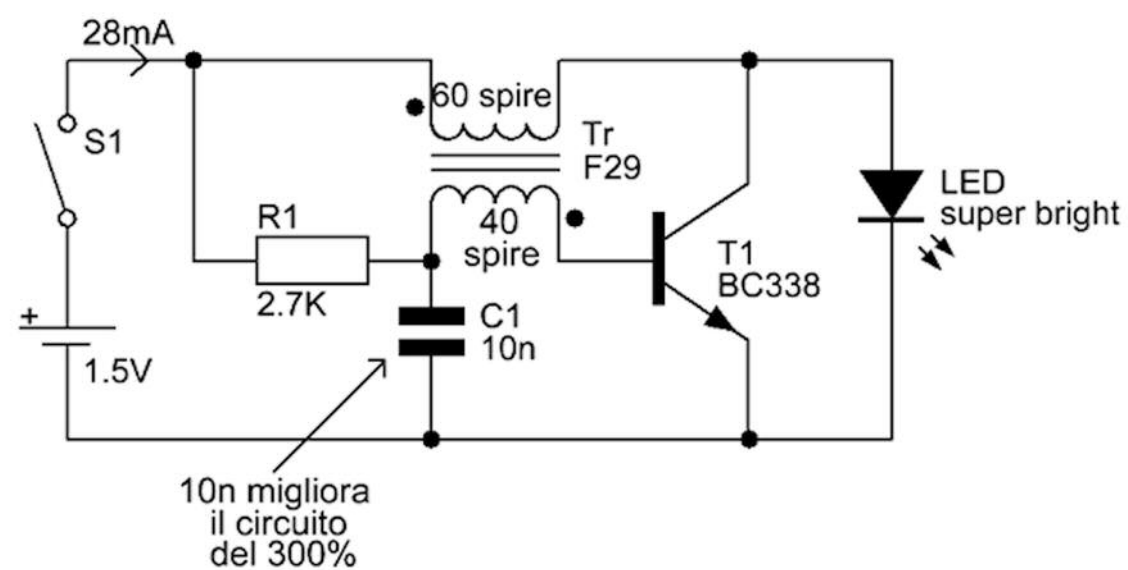



Figura 6: accensione di un LED bianco

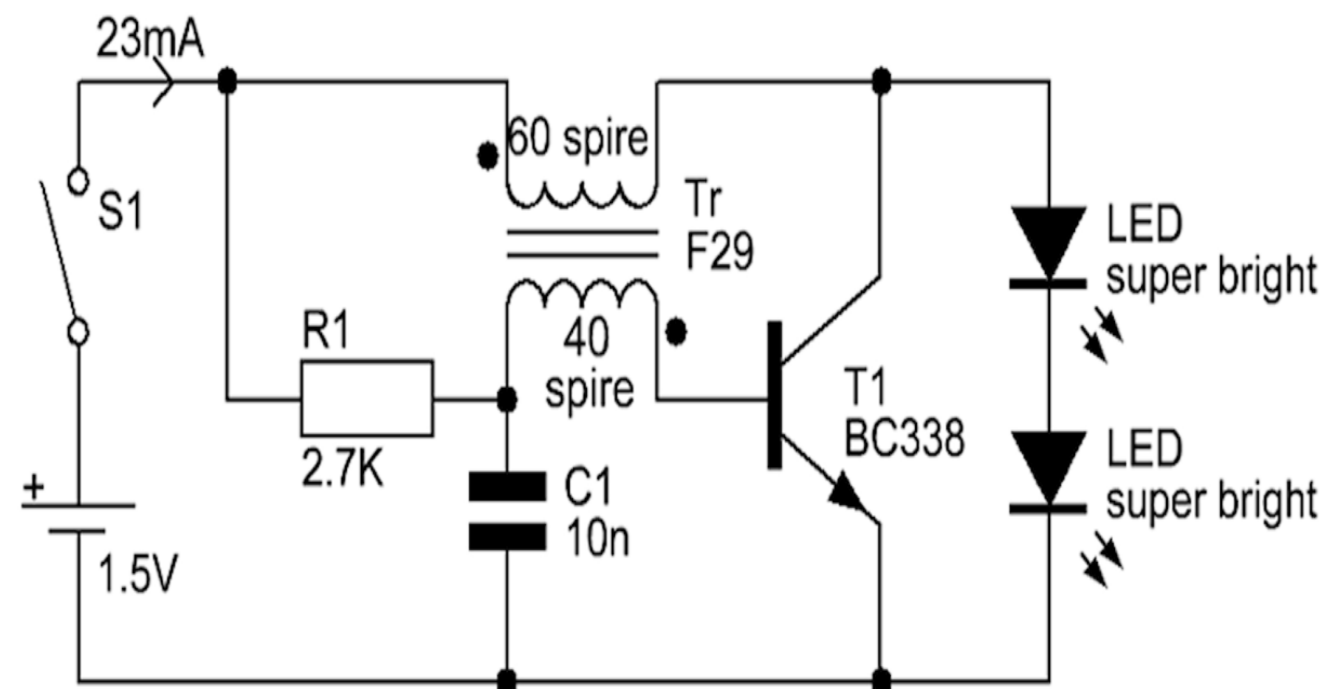



Figura 7: accensione di due LED bianchi

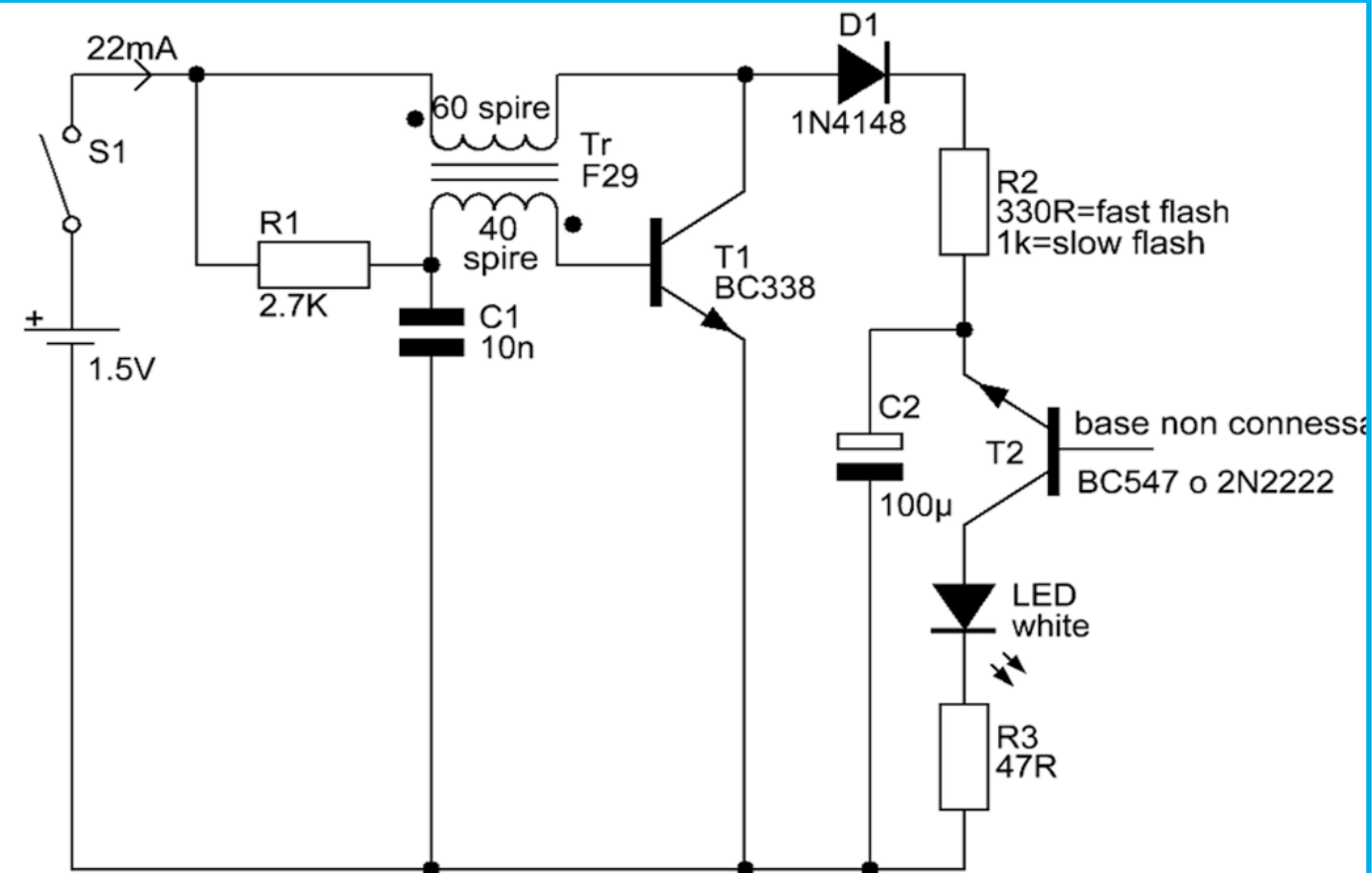



Figura 8: lampeggiatore con LED bianco

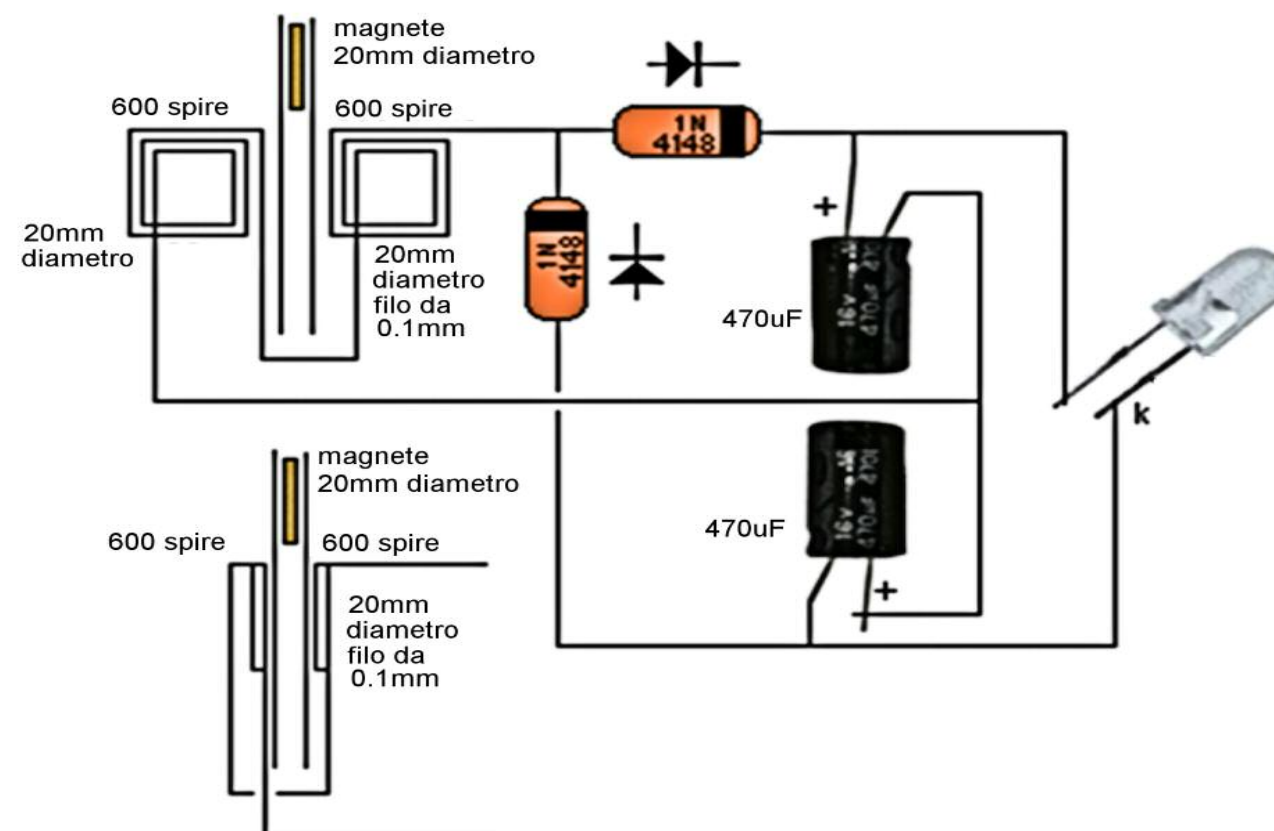



Figura 9: torcia a LED agitabile

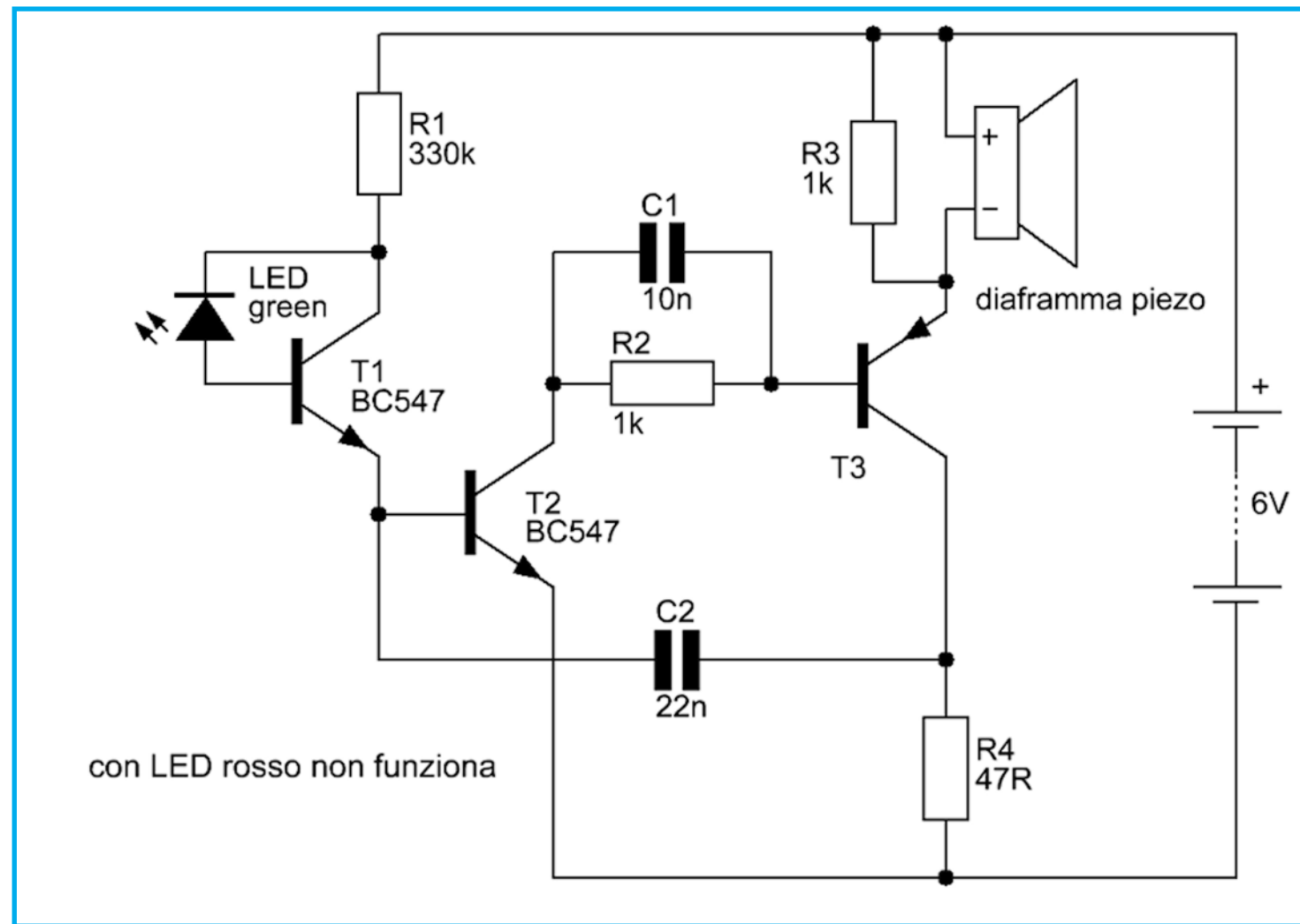



Figura 10: rilevatore di luce

Figura 11: rilevatore di cavi elettrici

LED BIANCO ALIMENTATO A 1,5V

Il circuito di figura 6 illumina un LED bianco usando un'unica pila.

DUE LED BIANCHI A 1,5V

Il circuito di figura 7 illumina due LED Bianchi usando un'unica pila.

LAMPEGGIATORE PER LED BIANCO

Il circuito di figura 8 fa lampeggiare un LED bianco usando una singola pila.

TORCIA A LED AGITABILE

Sullo schema di figura 9, sembra che gli avvolgimenti siano "appiattiti", mentre il magnete è appoggiato sul tavolo. Questo è solo uno schema per mostrare come le parti sono collegate. Le bobine sono in realtà appiattite vicino al cursore (contro il lato del magnete) come mostrato sullo schema. La tensione di uscita dipende da quanto velocemente il magnete passa da un estremo all'altro della corsa. Ecco perché un'agitazione più veloce produce una tensione più alta. Perché la tensione raggiunga il massimo è necessario che l'estremità del magnete abbia oltrepassato completamente le bobine. Questo è il motivo per cui la corsa eccede la posizione degli avvolgimenti sullo schema. Il circuito consiste di due avvolgimenti da 600 spire posti in serie, che pilotano un moltiplicatore di tensione x2.

Ogniqualvolta il magnete passa da un'estremità della corsa all'altra, le bobine producono un impulso positivo e uno negativo: l'impulso positivo carica il condensatore elettrolitico superiore attraverso il rispettivo diodo mentre quello negativo carica il condensatore inferiore attraverso il diodo corrispondente. Le tensioni attraverso ciascun elettrolitico vengono combinate per produrre l'alimentazione del LED bianco. Quando la tensione combinata è maggiore di 3,2V, il LED si illumina. Gli elettrolitici aiutano a mantenere il LED acceso mentre il magnete si appresta a eseguire un altro passaggio.

LED RILEVATORE DI LUCE

Il LED nel circuito di figura 10 rileverà della luce per attivare l'oscillatore. Un comune LED rosso non è adatto. Ma i LED verdi, gialli e i LED bianchi e rossi ad alta luminosità funzioneranno molto bene. La tensione di uscita del LED, quando è raggiunto da una luminosità molto alta, arriva a 600mV. Quando il LED rileva la luce, la sua resistenza diminuisce e una debole corrente scorre alla base del primo transistor. Questo amplifica la corrente di circa 200 volte e la resistenza tra collettore ed emettitore diminuisce. La resistenza da 330k sul collettore serve da limitatore, dato che il transistor intermedio richiede solo una debole corrente per far oscillare il circuito. Se la corrente è troppo elevata, il circuito si "bloccherà." Il trasduttore piezoelettrico non contiene componenti attivi e i toni che emette dipendono dal circuito che lo pilota.

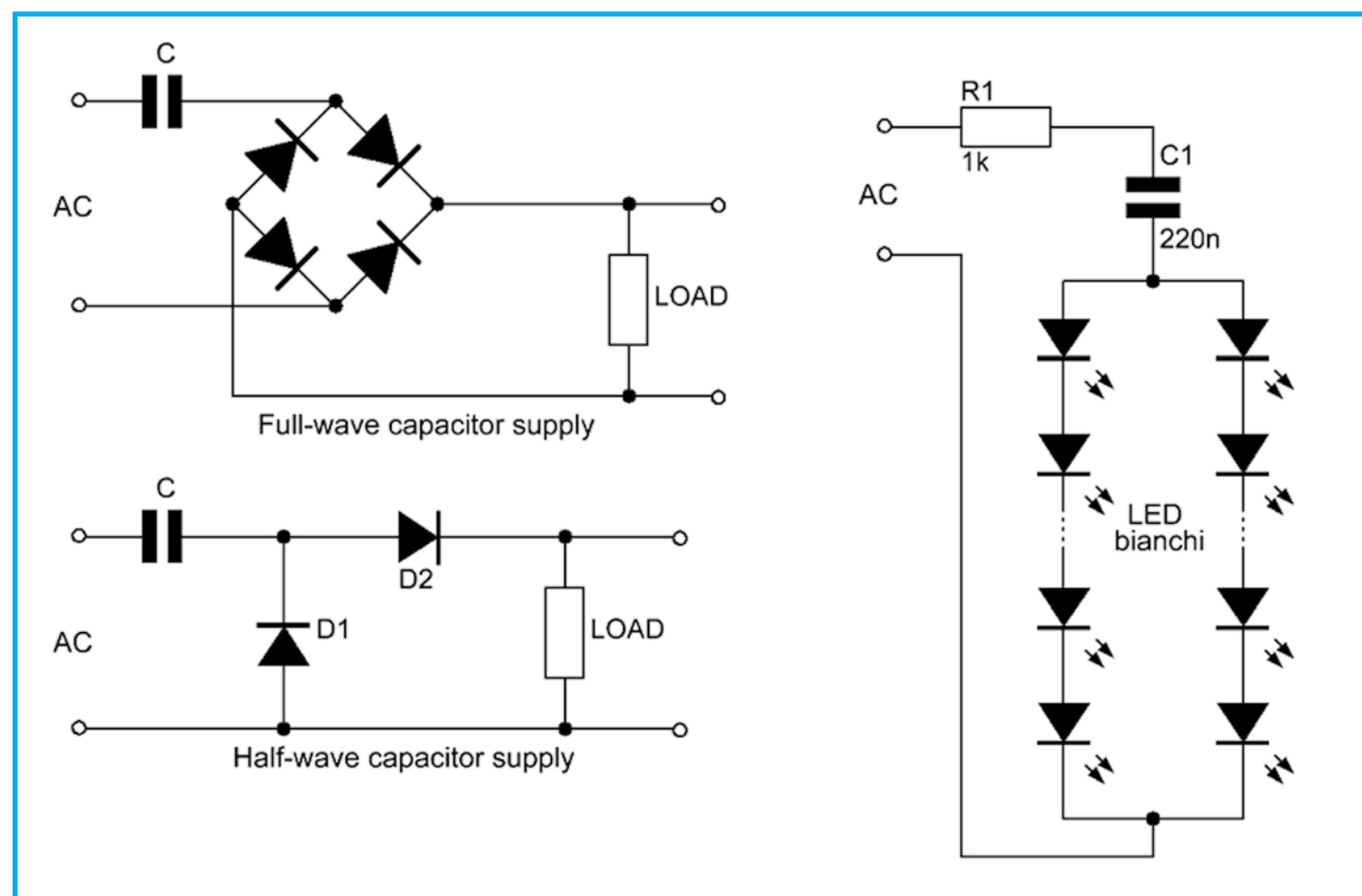



Figura 12: LED sui 240Vac

GUADAGNO DA 8 MILIONI!

Il circuito di figura 11 è così sensibile da rilevare "il ronzio di rete". Muovetelo semplicemente lungo le pareti e troverà per voi la posizione dei cavi. Ha un guadagno di circa 200 x 200 x 200 = 8.000.000 e rileverà anche l'elettricità statica e la presenza della vostra mano senza alcun contatto diretto. Sarete sorpresi da quanto riesce a trovare! L'elettricità statica è ovunque! L'ingresso di questo circuito si può considerare a impedenza molto elevata.

LED SU 240V

I circuiti collegati direttamente alla tensione di rete e di 240v sono da evitare per motivi di sicurezza. Tuttavia le luci degli alberi di Natale sono state collegate direttamente alla rete per oltre 30 anni senza grandi problemi. Si deve adottare un buon isolamento e le luci (LED) non devono essere a portata di mano. Vi servono almeno 50 LED per ogni stringa per evitare che vengano danneggiati da spunti eccessivi attraverso la resistenza da 1k - se il circuito viene acceso sul picco della forma d'onda. Aggiungendo ulteriori LED a ogni catena, la corrente avrà un leggero ribasso finché, al limite, quando raggiungerete i 90 LED per catena, si ridurrà a zero. Per 50 LED a catena, la tensione caratteristica totale sarà 180V quindi la tensione di picco sarà di 330V – 180V = 150V. Ciascun LED sopporterà un picco inferiore a 7mA durante il semi-periodo di illuminazione. La resistenza fa cadere 7V , dato che la corrente RMS è 7mA (7mA x 1.000 ohm = 7V). Non c'è bisogno di diodi raddrizzato-

C1
220n/400V
R1
1k
AC/220V
D1
1N4004
LED

C2
220n/400V
R2
1k
AC/220V
LED
LED

D1
1N4004
R1
10K / 3Watt
AC/220V
LED
white

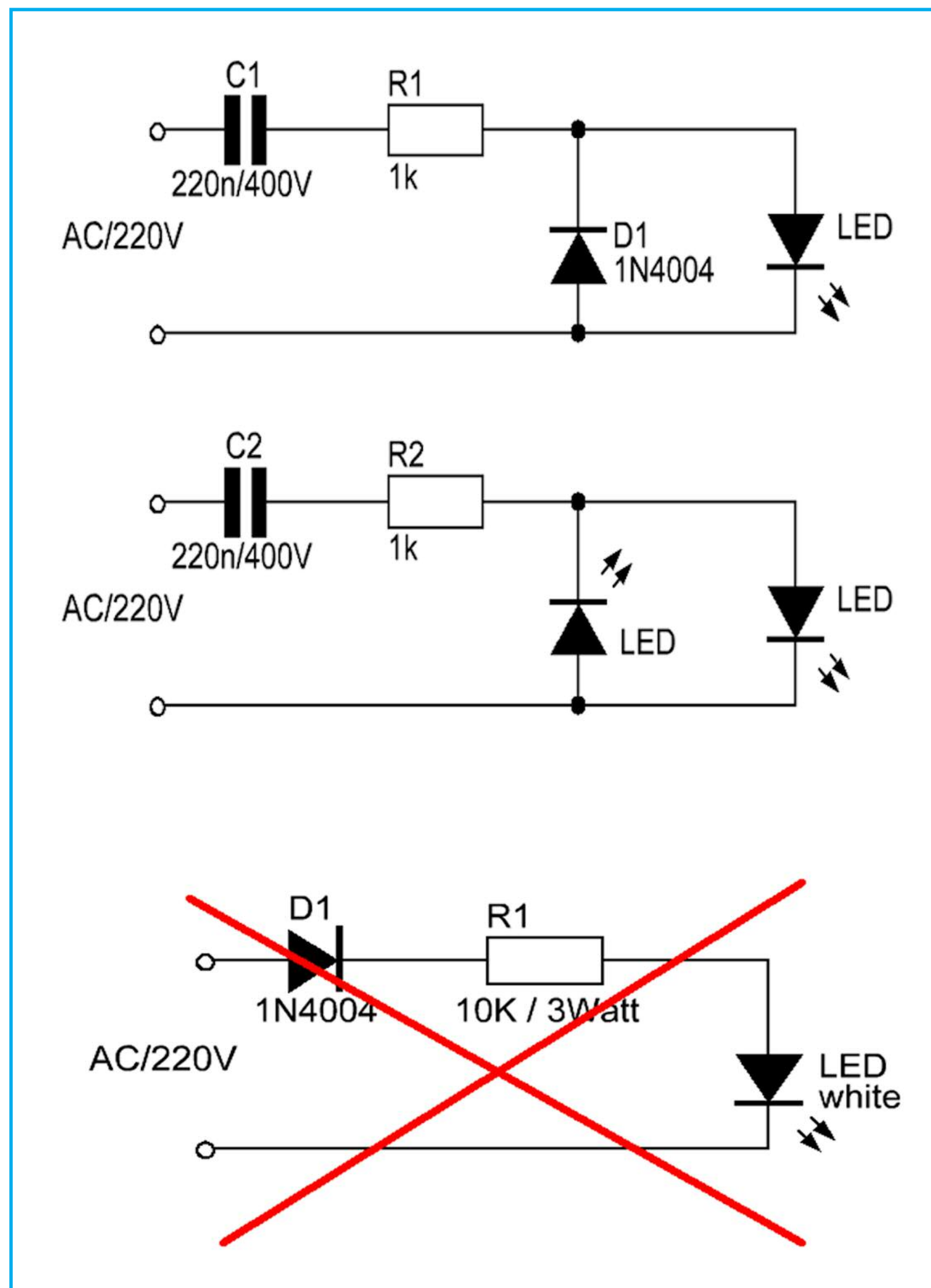



Figura 13: singolo LED sui 240Vac

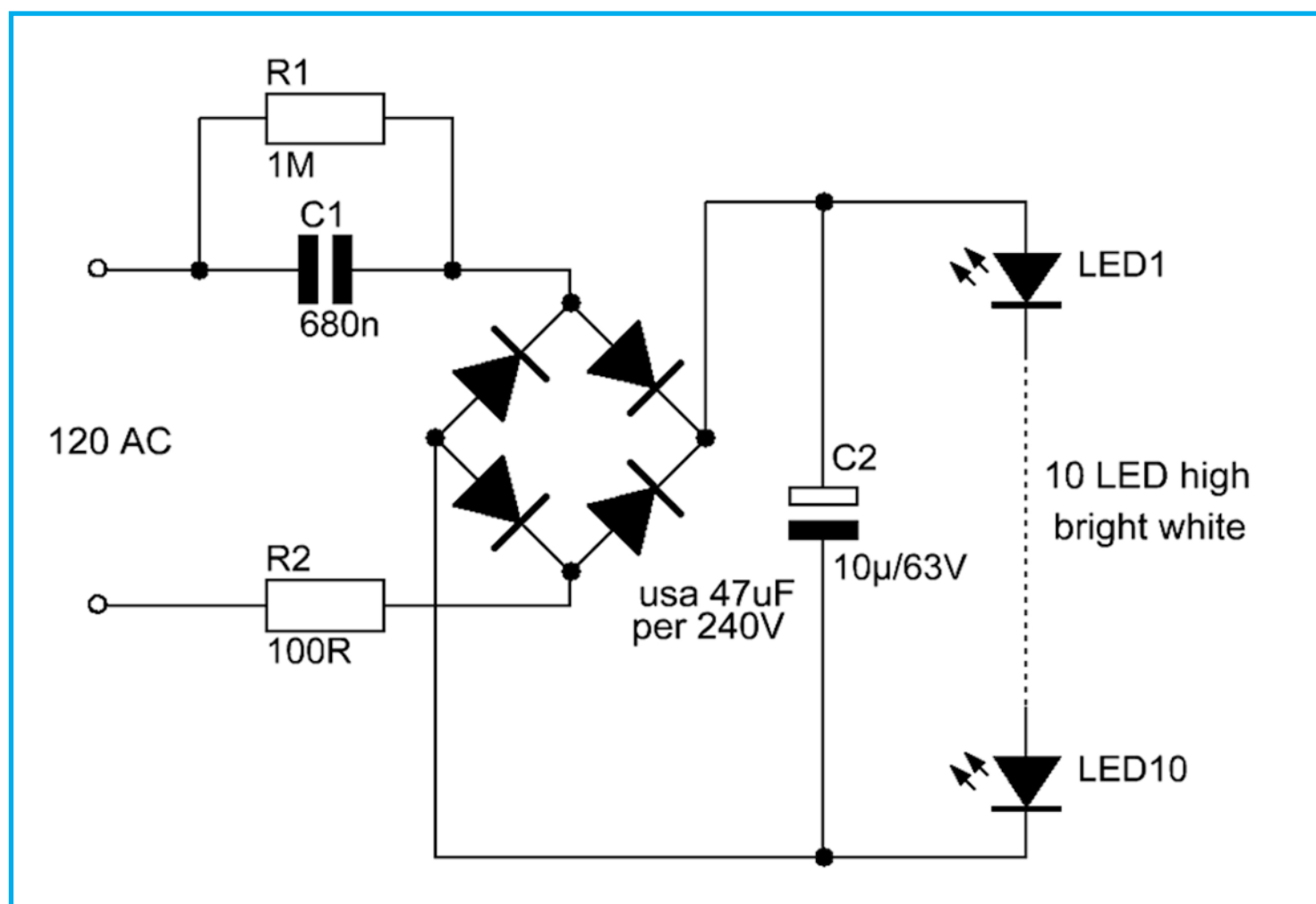



Figura 14: luce notturna

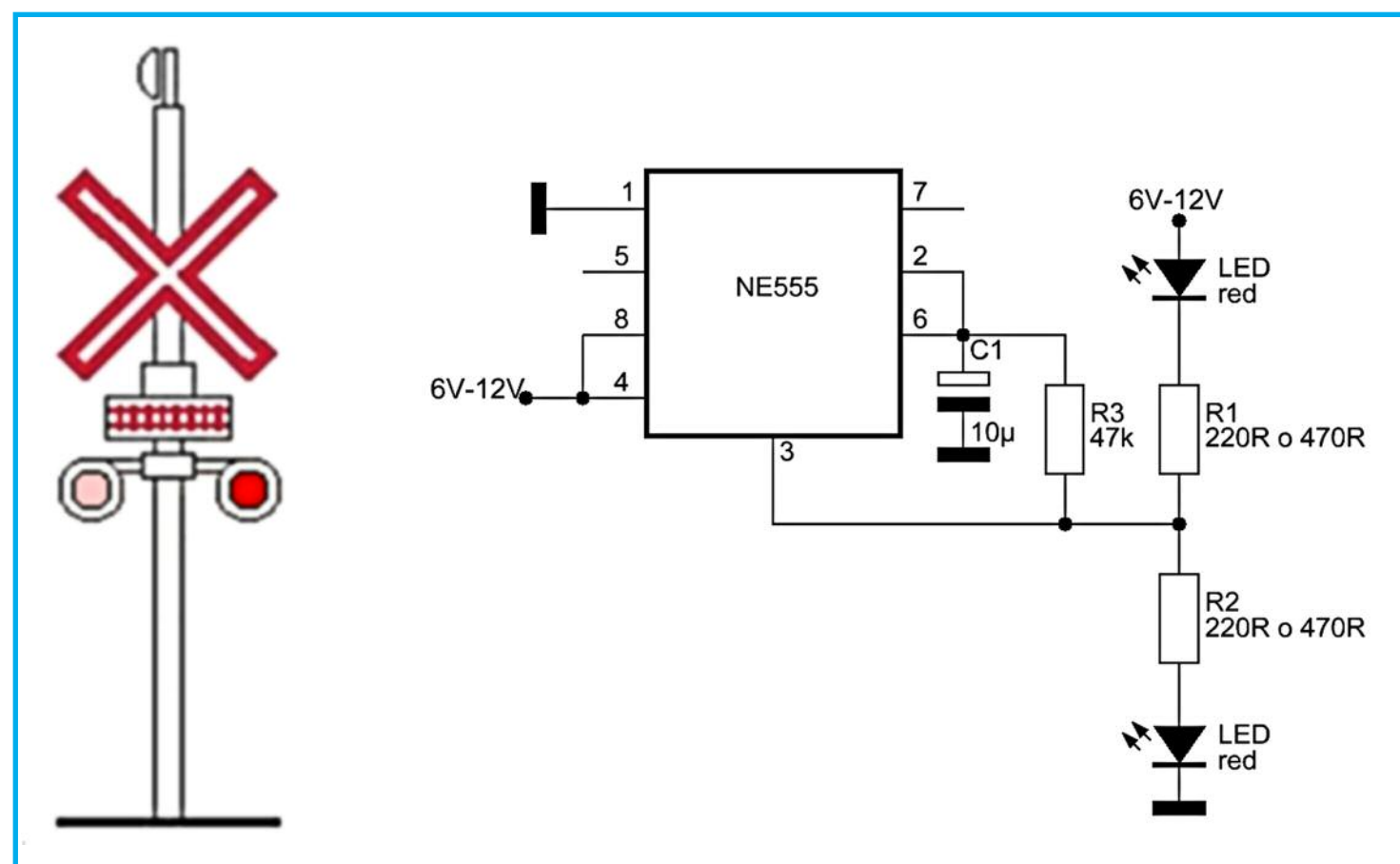



Figura 15: luci ferroviarie lampeggianti

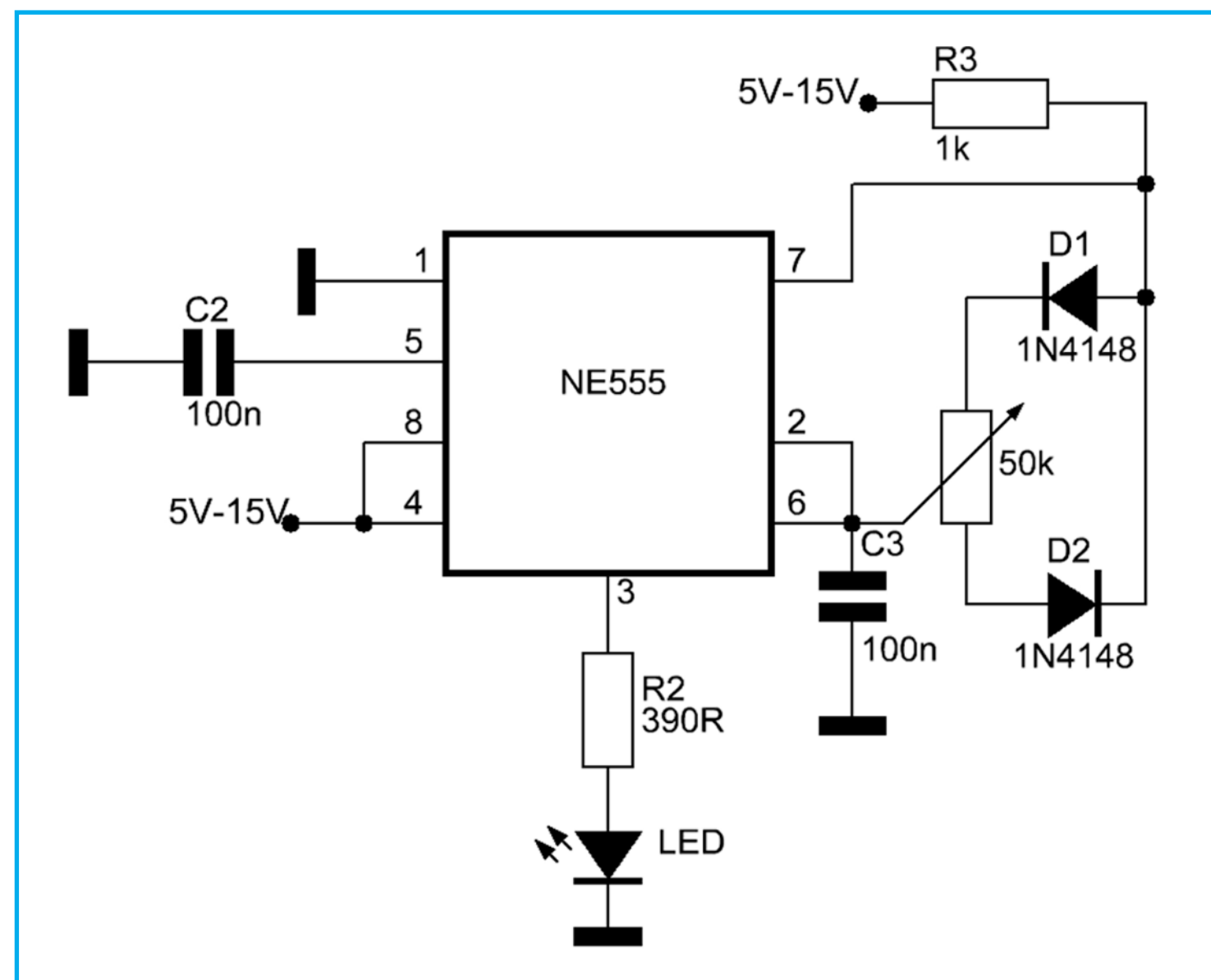



Figura 16: dimmer per LED

ri. La figura 12 mostra il collegamento dei LED. I LED sono i "raddrizzatori." Davvero intelligente.
Dovete avere LED in entrambe le direzioni per caricare e scaricare il condensatore. La resistenza serve soltanto a limitare alti picchi di corrente attraverso una delle catene di LED, nel caso in cui il circuito venga collegato quando la rete è al suo massimo. Questa può raggiungere i 330mA se si usa solo 1 LED, quindi il valore di questa resistenza deve essere adeguato se si impiegano LED in numero ridotto.
Un condensatore da 100n genererà 7mA RMS o 10mA di picco su onda intera o 3.5mA RMS (10mA di picco) sul semi-periodo. (con solo 1 LED per stringa). La capacità di corrente di un condensatore richiede ulteriori spiegazioni.
Sullo schema a sinistra vediamo un condensatore collegato a un alimentatore a onda intera. Questo è esattamente lo stesso caso del circuito LED a 240v illustrato prima. Immaginate di rimuovere la resistenza LOAD (sostituendola con un corto). Due diodi saranno rivolti verso l'alto e due verso il basso. Questa è esattamente la stessa situazione che abbiamo trovato nel circuito precedente, con la differenza che il punto intermedio è unito. Ciò significa che

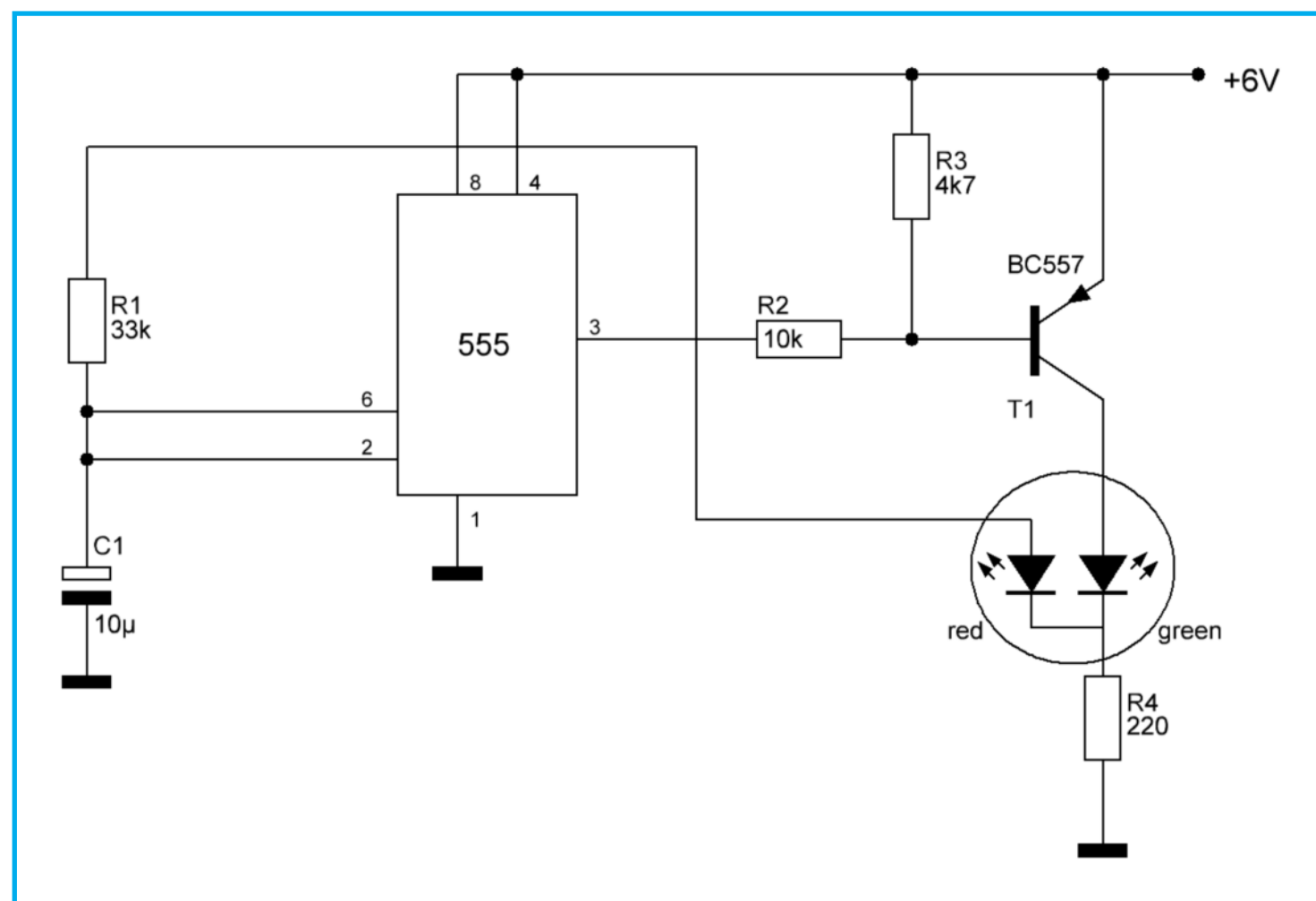



Figura 17: pilotaggio di un LED bicolore

Figura 18: pilotaggio di un LED bipolare

ogni 100n di capacità genereranno 7mA RMS (10mA di picco su ciascuna semi-onda).
Nell'alimentatore a semi-onda, il condensatore genera sul carico 3.5mA RMS per ogni 100n (10mA di picco sul ciascun semi-periodo, ma uno dei semi-cicli viene perso nel diodo), e durante l'altro semi-periodo il picco di 10mA viene perso nel diodo che scarica il condensatore.
Potete usare qualsiasi tipo di LED e cercare di mantenere uguale la caduta di tensione in ciascuna stringa. Di fatto ognuna di esse lavora in DC. Non si tratta di DC costante ma di DC pulsante.
Infatti la corrente è a zero per 1/2 ciclo, poi resta nulla finché la tensione sale oltre la tensione caratteristica complessiva di tutti i LED, quindi un aumento graduale della corrente per la parte rimanente del ciclo, quindi una discesa graduale verso zero sul fronte discendente dell'onda, quindi nulla per 1/2 ciclo. Dato che i LED si accendono e si spengono, potreste notare dei tremolii, questo è il motivo per cui le due stringhe devono essere sistemate vicine.

LED SINGOLO SU 240V

Un singolo LED può essere illuminato utilizzando un condensatore da 100n o 220n con tensione di lavoro di 400V. Questi condensatori sono definiti "X2" e progettati per il collegamento diretto alla rete. In figura 13 lo schema di collegamento. All'inversione tra neutro e fase i LED si troveranno a 240V rispetto a terra e questo rappresenta una scarica da oltre 340v in caso di contatto.
Il diodo di potenza sul primo schema ha il compito di scaricare il condensatore da 0.22u durante un semi periodo, di modo che si possa caricare nell'altro e inviare energia al LED. La resistenza da 1k limita il picco di corrente, se all'accensione la rete è al suo massimo. Allo stesso modo si possono pilotare due LED con il medesimo circuito, dato che un LED si illuminerà durante un semi-periodo e l'altro LED nel semi-periodo successivo.

LUCE NOTTURNA DA RETE

Il circuito di figura 14 illumina una serie di 10 LED bianchi. Il condensatore da 10u e la resistenza 100R riducono le variazioni di luce.

LUCI FERROVIARIE LAMPEGGIANTI

Il circuito di figura 15 fa lampeggiare due LED alternativamente. E' ideale per un plastico nel controllo delle luci dei passaggi a livello ferroviari.

LED DIMMER

Per regolare la luminosità di uno o più LED dal 5% al 95% si può utilizzare il circuito di figura 16. Il circuito è sostanzialmente un oscillatore, o meglio un multivibratore astabile, che accende e spegne il LED a diverse frequenze. Il risultato per l'occhio umano è la diversa percezione della luminosità. Ovviamente se le frequenza diviene

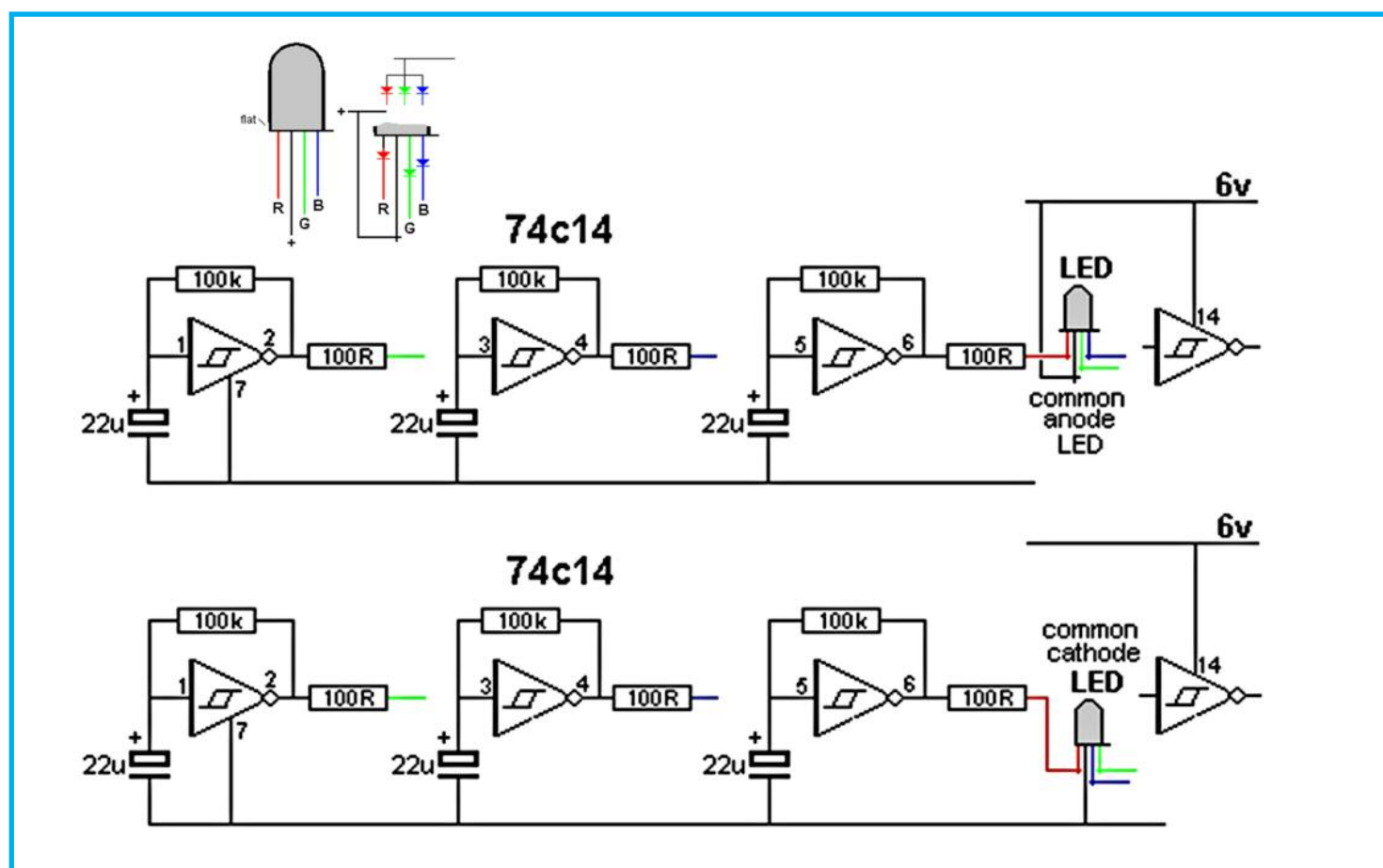



Figura 19: pilotaggio di LED RGB

troppo bassa, si potrà percepire l'effettivo lampeggio del LED.

PILOTARE UN LED BI-COLORE

Alcuni LED a tre terminali producono luce rossa e verde. Il circuito di figura 17 fa lampeggiare alternativamente un LED bicolore rosso-verde. Il circuito è basato su un timer 555 la cui uscita pilota direttamente il led rosso, mentre il led verde viene pilotato attraverso il transistore che funziona da invertitore.

PILOTARE UN LED BI-POLARE

Alcuni LED a due terminali emettono luce rossa e verde a seconda della polarità della tensione ad essi applicata. Sono definiti LED Bi-polari. Il circuito di figura 18 accende alternativamente un LED bi-polare rosso e verde.

PILOTARE UN LED RGB

In figura 19 un semplice circuito di pilotaggio che gestisce 3 LED in un LED RGB per produrre una varietà di colori interessanti.

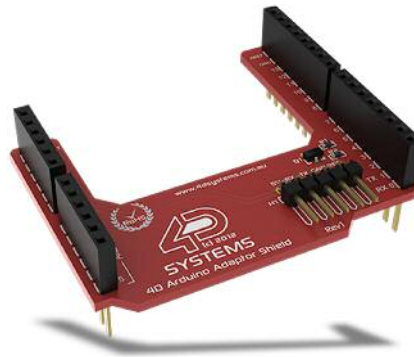
Anche se il valore dei componenti è identico nei tre oscillatori, la leggera differenza nelle tolleranze genererà una sequenza casuale di colori che non si ripeterà nell'immediato.

I colori cambiano repentinamente, dato che il circuito non usa la "Pulse Width Modulation" per produrre una variazione graduale dall'uno all'altro. Questo LED è definito ad ANODO COMUNE. Questa scelta consente il collegamento a transistor o altri dispositivi verso terra. Il secondo circuito usa un LED a catodo comune. Notate il collegamento diverso.

IEshop

SHIELD ADATTATORE PER I MODULI 4D SYSTEM

Il 4D Arduino Adaptor Shield è un semplice Shield per Arduino che permette ad Arduino di connettersi , tramite un'interfaccia seriale in un fattore di forma comoda , a molti moduli display di 4D System con una singola connessione a 5 pin.

Il 4D Arduino Adaptor Shield è compatibile con molti shield popolari, e utilizza solo un pin digitale per la connessione con Arduino

Prezzo: € 8.54

PIC CLICKER

PIC clicker è un kit di sviluppo incredibilmente compatto che porta l'innovativo socket mikroBUS al vostro microcontrollore preferito, collegando la scheda Click desiderata per valorizzare il vostro progetto con caratteristiche impressionanti. La PIC Clicker è preprogrammata con un veloce bootloader USB HID e usandolo per caricare il firmware, la PIC Clicker diventerà subito un dispositivo pronto per lavorare.

La scheda dispone di tutto ciò che serve per iniziare: il microcontrollore PIC18F47J53, connettore USB, due LED e pulsanti, pulsante di reset, connettore mikroProg e header per l'interfacciamento con l'elettronica esterna.

Prezzo: € 16.47

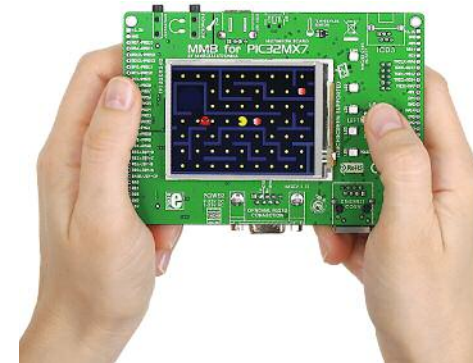
BASIC STAMP DISCOVERY KIT (USB)

Il nostro microcontrollore BASIC Stamp più popolare è certamente il modulo BS2-IC. Largamente usato nell'istruzione, hobby e industria, questo modulo ha abbastanza spazio programma, velocità d'esecuzione e pins I/O per molte applicazioni. Combinato con la nostra piattaforma Board of Education e il What's a Microcontroller Parts and Text Kit, avete un kit "Getting Started" per la programmazione e sperimentazione dei microcontrollori.

Prezzo: € ~~169.58~~ € 120.78

MEMBERSHIP CLUB FULL

Fare Elettronica è una rivista di elettronica presente sul mercato italiano da quasi trent'anni. Il target dei lettori è suddiviso in progettisti elettronici, docenti e studenti di scuole ed università ad indirizzo elettronico. I contenuti sono divisi in Approfondimenti, Progetti ed Autocostruzione, Tutorials, Robotica, Radiofrequenza ed altre Rubriche. Firmware è l'unica rivista italiana (e probabilmente europea) completamente dedicata ai microcontrollori, dispositivi FPGA e componentistica analogica. Il target dei lettori è quello dei progettisti elettronici. I contenuti sono divisi in: Approfondimenti sulle tecnologie, Tecniche di programmazione, Strumenti di sviluppo, Analogica ed altre Rubriche. La CLUB Membership comprende l'accesso e/o download agli articoli (centinaia) del CLUB di Fare Elettronica e Firmware

Prezzo: € 19.99

MULTIMEDIA FOR PIC32MX7

Questa scheda fornisce una piattaforma di sviluppo multimediale d'alta qualità completa per il dispositivo PIC32MX7. Possiede numerosi moduli on-board che vi permettono di scrivere applicazioni multimediali complesse. Questa scheda può essere usata sia per lo sviluppo o come prodotto finale. Il microcontrollore PIC32MX795F512L on-board è pre-programmato con un veloce bootloader USB e, quindi, pronto a lavorare in brevissimo tempo. Non dovete spendere un euro in più sui programmatori esterni. Per coloro i quali ne necessitano, la scheda può, inoltre, essere programmata utilizzando un programmatore e debugger esterno mikroProg o usando un altro programmatore compatibile con il connettore ICD2/3.

Prezzo: € 131.76

EASYAVR V7

EasyAVR v7 è la settima generazione della nostra famosa scheda di sviluppo AVR. E' il prodotto di conoscenze accumulate nel corso degli ultimi 10 anni, ed è lo stato dell'arte del design, funzionalità e qualità. E' amata da principianti e hobbisti, così come dai professionisti, ed è utilizzata nella formazione e nello sviluppo da migliaia di persone. La scheda viene fornita con ATMEGA32.

Prezzo: € 132.86

DEMO KIT D868TINY-LITE

Il demokit comprende 2 moduli TinyOne Lite, 2 Mother board RS232, 2 Cavi seriali, 2 alimentatori 12V 300mA, 2 batterie alcaline 9V 6LR61, 2 antenne e un Cd-Rom con tutta la documentazione. Il modulo TinyOne lavora su Frequenza di 868 Mhz con una potenza di trasmisisone fino a 10mW che garantisce una range fino a 500m. La trasmissione è multicanale con Radio Data rate fino a 38.4Kbps.

Prezzo: € 305.00
SENSORE GAS

La scheda Gas Sensor è progettata per funzionare in combinazione con uno dei sensori di gas elencati di seguito (non incluso) che consente di determinare quando un livello, reimpostato, di gas è stato raggiunto o superato.

-Modulo Sensore Gas CO (monossido di carbonio)
-Modulo Senosre Gas CH4 (metano)
-Modulo Sensore Gas GPL (propano)
-Modulo Sensore Gas C2H5Oh (alcol)

Il sensore di gas si connette semplicemente alla scheda Gas Sensor, il che rende facile da usare questi sensori in progetti embedded. La scheda può interfacciarsi ad un microcontrollore con solo due pin I/O, consentendo al sensore di essere controllato e monitorato automaticamente.

Prezzo: € 23.18

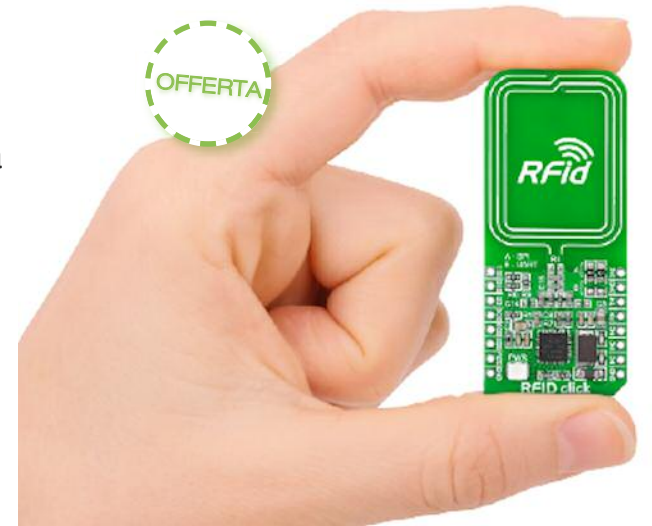
ARDUINO ESPLORA

Arduino Esplora è la nuova scheda Arduino pronta all'uso e di facile utilizzo con cui esplorare le infinite possibilità del mondo dei sensori e degli attuatori, senza dovere avere a che fare con breadboard, saldature e cavi. Non c'é limite alle applicazioni possibili che si possono realizzare e all'ampia fornitura di sensori, è possibile aggiungerne altri all'occorrenza. Aggiungendo un modulo LCD sarai in grado di creare un video game originale sulla tua personale console open-source. Arduino Esplora combina il processore Arduino con una gamma di sensori e attuatori.

NOVITÀ

Prezzo: € 50.02

RFID CLICK

RFid Click una scheda accessoria che utilizza il nuovo socket mikroBUS. Si tratta di una soluzione compatta e semplice per aggiungere la tecnologia RFID al vostro progetto . È dotato di un transceiver contactless a 13,56 MHz (CR95HF), così come una trace antenna. RFid Click comunica con il microcontrollore del la scheda di destinazione via mikroBUS UART (TX, RX), SPI (MISO, MOSI, SCK, CS) INT, RST, PWM e linee AN . La scheda è stata progettata per usare solo l'alimentazione a 3.3V. Un diodo LED (verde) indica la presenza di alimentazione. L'IC CR95HF è un modulo RFID con transceiver integrato per applicazioni contactless.

Prezzo: € 23.79

STARTER KIT PER 4DLCD-FT843 CON SHIELD PER ARDUINO

Per qualsiasi progetto che richiede la funzione dell'FT800 Video Engine, un Display TFT 4.3"e Arduino.
4D Systems ha progettato un Arduino Shield soprannominato ADAM (Arduino Modulo display Adaptor) per interfacciare il display 4DLCD-FT843, guidato dall'FTDI FT800 Video Engine, ad Arduino.
L'ADAM è progettato per interfacciarsi con il Display 4DLCD - FT843, e permette il collegamento diretto tra un Arduino e il 4DLCD - FT843 SPI Display, che è alimentato da FTDI FT800 Video Engine. Il display è collegato tramite un connettore FPC a 10 vie. ADAM è alimentato dal bus 5V dell'Arduino , che poi regola questo a 3,3 V per il display 4DLCD - FT843.

Prezzo: € 67.10

DISPLAY EMBEDDED SPI

Il 4DLCD-FT843 è un display SPI embedded di 4D System, con un display touch resistivo da 4,3" e con integrato il controller grafico FTDI FT800 Video Engine.
Il 4DLCD-FT843 è un potente display SPI che consente ad un host SPI, dunque una interfaccia ad alta velocità, di essere collegato direttamente al display, e fornisce un potente set di funzionalità grafiche per l'host utilizzando la piattaforma EVE (Embedded Video Engine) il chip FTDI FT800 Video Engine.

Questo display fornisce quattro funzionalità con un solo dispositivo: un Display TFT LCD 4.3”, Audio con ingresso abilitazione amplificatore, Tousch Screen resistivo 4-wire e un motore integrato Video Engine. Queste combinazioni costruite direttamente nello schermo, permettono una semplice integrazione in tantissime applicazioni.
Il processore FT800 esegue la versione FTDI di EVE (Embedded Video Engine: un controller grafico dedicato a display grafici di alta qualità con il supporto Widget ed è progettato per scaricare il processore host e fornire una serie di funzionalità grafiche).

 Prezzo: € 47.58

MODULO BLUETOOTH 2.0

F2M03GXA-S1 è un modulo embedded Bluetooth 2.0+ EDR in classe 1 con eccezionali carateristiche di trasmissione wirless, a bordo una efficente antenna omni-direzionale e profilo serial port (SPP). E’ completamente compatibile con dispositivi Bluetooth per comunicazioni dati e voce. Il modulo ha una potenza di trasmissione fino a +19dBm e una sensibilità di ricezione fino a -92dBm. Tutte le informazioni inviate alla porta seriale vengono trasmesse in modo trasparente via Bluetooth al dispositivo collegato. F2M03GXA è inoltre disponibile con Headset, Hands Free o HCI firmware. Firmware opzionali: DUN, AGP, OBEX, HID o HCRP.

 Prezzo: € 31.72

- Ordina merce per soli 450 Eur (IVA e spese di trasporto escluse) dal 12.12.13 al 31.3.14 e ricevi un buono vacanza per 2 adulti e un bambino. Clicca per saperne di più.
- Recensisci i prodotti acquistati e ti scontiamo 5 Eur sul tuo prossimo acquisto!
- Diventa membro IE Cloud e riceverai offerte a te dedicate!